

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

**AN ANALYSIS OF FUEL EFFICIENCY AS A KEY
PERFORMANCE PARAMETER FOR MILITARY
AIRCRAFT ACQUISITIONS**

by

Laurence M. Young

September 2002

Thesis Advisor:

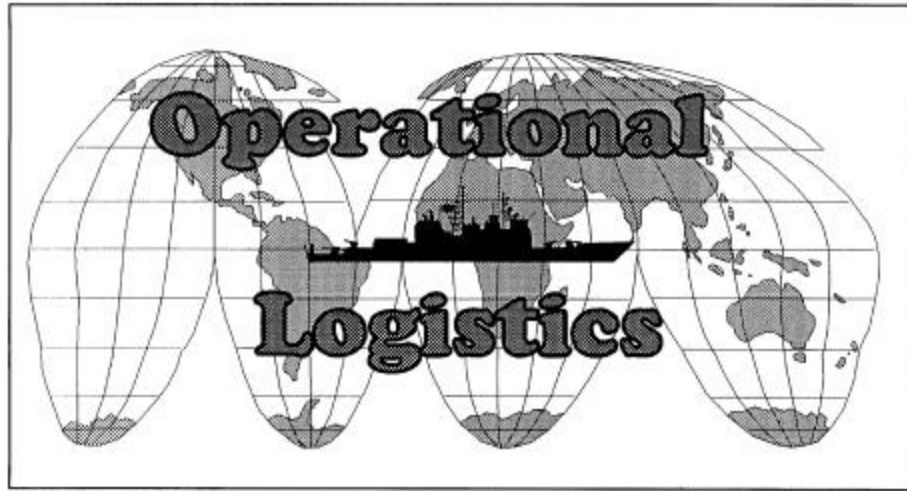
Lyn R. Whitaker

Second Reader:

Paul J. Sanchez

Approved for public release; distribution is unlimited

*Amateurs discuss strategy,
Professionals study logistics*



REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2002	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: An Analysis of Fuel Efficiency as a Key Performance Parameter for Military Aircraft Acquisitions.			5. FUNDING NUMBERS	
6. AUTHOR(S) Young, Laurence Martin				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The Acquisition Management System (AMS) of the Department of Defense (DoD) is established to ensure that the needs of the warfighter are fulfilled in the most cost-effective manner possible. The establishment of Key Performance Parameters (KPPs) on Operational Requirements Documents (ORDs) ensures that critical measures of performance are continuously monitored as the identification of a military requirement becomes a fulfilled capability. Threshold values for KPPs identified during an acquisition determine the minimum acceptable level of performance. Failure to meet the threshold level of an individual KPP is grounds for program cancellation. The Chairman of the Joint Chiefs of Staff (CJCS) has mandated that all defense acquisition programs should be kept to a maximum of eight for any individual program. Due to the multi-faceted aspect of modern military aircraft acquisition programs, frequently there are numerous parameters of critical importance. For this reason, any repetition of measure between KPPs must be minimized. The decision of the Principal Deputy Undersecretary of Defense for Acquisition, Technology and Logistics (PUSD (AT&L)) to include fuel efficiency as a KPP on all future acquisitions is an unnecessary exercise for military aircraft. This thesis develops a model in which multiple aircraft are examined flying different mission profiles carrying different weapon payloads. Strong relationships between factors such as combat radius, payload and on-station endurance versus fuel efficiency are demonstrated by regression analysis of the model output. Consequently, any measure of fuel efficiency as a KPP is shown to be redundant. Furthermore, the methodology developed in this thesis may have further uses for the development of logistics planning factors for present and future military aircraft applications.				
14. SUBJECT TERMS Fuel Efficiency, Key Performance Parameters			15. NUMBER OF PAGES 130	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**AN ANALYSIS OF FUEL EFFICIENCY AS A KEY PERFORMANCE
PARAMETER FOR MILITARY AIRCRAFT ACQUISITIONS**

Laurence M. Young
Lieutenant, United States Navy
B.S., United States Naval Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

**NAVAL POSTGRADUATE SCHOOL
September 2002**

Author: Laurence M. Young

Approved by: Lyn R. Whitaker
Thesis Advisor

Paul J. Sanchez
Second Reader

James N. Eagle, Chairman
Operations Research Department

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Acquisition Management System (AMS) of the Department of Defense (DoD) is established to ensure that the needs of the warfighter are fulfilled in the most cost-effective manner possible. The establishment of Key Performance Parameters (KPPs) on Operational Requirements Documents (ORDs) ensures that critical measures of performance are continuously monitored as the identification of a military requirement becomes a fulfilled capability. Threshold values for KPPs identified during an acquisition determine the minimum acceptable level of performance. Failure to meet the threshold level of an individual KPP is grounds for program cancellation. The Chairman of the Joint Chiefs of Staff (CJCS) has mandated that all defense acquisition programs should be kept to a maximum of eight for any individual program. Due to the multifaceted aspect of modern military aircraft acquisition programs, frequently there are numerous parameters of critical importance. For this reason, any repetition of measure between KPPs must be minimized. The decision of the Principal Deputy Undersecretary of Defense for Acquisition, Technology and Logistics (PUSD (AT&L)) to include fuel efficiency as a KPP on all future acquisitions is an unnecessary exercise for military aircraft. This thesis develops a model in which multiple aircraft are examined flying different mission profiles carrying different weapon payloads. Strong relationships between factors such as combat radius, payload and on-station endurance versus fuel efficiency are demonstrated by regression analysis of the model output. Consequently, any measure of fuel efficiency as a KPP is shown to be redundant. Furthermore, the methodology developed in this thesis may have further uses for the development of logistics planning factors for present and future military aircraft applications.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	OVERVIEW.....	1
B.	BACKGROUND.....	2
C.	OBJECTIVE STATEMENT.....	5
D.	OVERVIEW.....	6
II.	THE ACQUISITION PROCESS.....	9
A.	ACQUISITION PROGRAMS.....	9
1.	ACAT I Programs.....	9
2.	Other Acquisition Programs.....	10
B.	ACQUISITION RESPONSIBILITIES.....	10
1.	Joint Requirements Oversight Council (JROC).....	11
2.	Defense Acquisition Board (DAB).....	11
3.	Defense Resources Board (DRB).....	12
4.	Deputy Secretary of Defense (DepSecDef).....	12
5.	Under Secretary of Defense for Acquisition, Technology and Logistics (USD (AT&L)).....	12
6.	Chairman of the Joint Chiefs of Staff (CJCS).....	12
C.	ACQUISITION DOCUMENTS.....	12
1.	Mission Need Statement (MNS).....	12
2.	Operational Requirements Document (ORD).....	12
3.	Key Performance Parameters (KPPs).....	13
4.	Thresholds and Objectives.....	13
5.	Acquisition Program Baseline (APB).....	13
III.	MODEL DESCRIPTION.....	15
A.	INITIAL SAC CHART ANALYSIS.....	15
B.	JAVA MODEL.....	20
1.	Java Method: climb.....	22
2.	Java Methods: loiter, speedyTime, cruise and speedyDist.....	24
3.	Aerodynamic Drag.....	29
a.	<i>Drag for the SH-60B and HH-60H Seahawk Helicopters....</i>	<i>29</i>
b.	<i>Drag for the CH-53E Super Stallion Helicopter.....</i>	<i>29</i>
c.	<i>Drag for the AH-1W Cobra and AH-64A Apache Helicopters.....</i>	<i>31</i>
d.	<i>Drag for the F-14A Tomcat and F/A-18C Hornet.....</i>	<i>32</i>
IV.	MISSION PROFILES AND REGRESSION ANALYSIS.....	35
A.	AH-1W COBRA MISSION PROFILE.....	35
B.	AH-64A APACHE MISSION PROFILE.....	39
C.	CH-53E SUPER STALLION MISSION PROFILE.....	41
D.	F-14A TOMCAT MISSION PROFILE.....	44
E.	F/A-18C HORNET MISSION PROFILES.....	47

1.	TestHornet2 Mission Profile	48
2.	TestHornet4 Mission Profile	50
F.	SEAHAWK HELICOPTER MISSION PROFILES	52
1.	TestSeahawk1 and TestSeahawk2 Mission Profiles	53
2.	TestSeahawk3 and TestSeahawk4 Mission Profiles	59
V.	CONCLUSIONS AND RECOMMENDATIONS.....	63
A.	SURROGATES FOR FUEL EFFICIENCY	63
B.	A MODEL FOR LOGISTICS PLANNING FACTORS.....	64
	APPENDIX A. STANDARD ATMOSPHERE	67
	APPENDIX B. AIRCRAFT CLASS JAVADOC CONSTRUCTOR AND METHOD	
	SUMMARIES.....	69
A.	INTERFACE AIRCRAFT.....	69
B.	CLASS AH1W.....	71
C.	CLASS AH64A.....	76
D.	CLASS CH53E	88
E.	CLASS F14AStd.....	93
F.	CLASS FA18CStd.....	95
G.	CLASS HH60H	97
H.	CLASS SH60B.....	99
	LIST OF REFERENCES	103
	INITIAL DISTRIBUTION LIST	109

LIST OF FIGURES

Figure 1.	U.S. Dependence on Foreign Oil (From: DSB 2001).....	1
Figure 2.	How an Idea Becomes a Capability (From: NWC 2002)	3
Figure 3.	KPPs from Recent ACAT1 Aviation Programs (From: JHU APL 2002)	6
Figure 4.	Acquisition Oversight (From: NWC 2002)	11
Figure 5.	Typical SAC Chart Cover Sheet (From: NAVAIR 1970)	15
Figure 6.	Excerpt T-2B SAC p. 4 Performance Summary (From: NAVAIR 1970).....	16
Figure 7.	Excerpt T-2B SAC p. 8 Mission Profiles (From: NAVAIR 1970).....	18
Figure 8.	F-14D SAC Chart Burn Rate Versus Combat Radius and Payload.....	19
Figure 9.	F/A-18C SAC Chart Burn Rate Versus Combat Radius and Payload	19
Figure 10.	AH-1W Climb Performance (From: NAVAIR 2001)	23
Figure 11.	HH-60H Level Flight Performance (NAVAIR 2001)	25
Figure 12.	F/A-18C Level Flight Performance (From: NAVAIR 2001)	27
Figure 13.	CH-53E External Load Drag (From NAVAIR 2001).....	30
Figure 14.	CH53E External Load Flight Performance (From: NAVAIR 2001)	31
Figure 15.	AH-1W and AH-64A Weapons Stations	32
Figure 16.	F-14A Sample Drag Computation (From: NAVAIR 2001)	32
Figure 17.	F/A-18C Sample Drag Computation (From: NAVAIR 2001)	34
Figure 18.	AH-1W Mission Profile	35
Figure 19.	Actual Versus Fitted Values	37
Figure 20.	Residual Plot for TestCobra1.java	38
Figure 21.	AH-64A Mission Profile	39
Figure 22.	Actual versus Fitted values for TestApache1	41
Figure 23.	CH53E Mission Profile	41
Figure 24.	CH-53E External Cargo	42
Figure 25.	Actual Versus Fitted Values for TestSeaStallion2	44
Figure 26.	F-14A Mission Profile	44
Figure 27.	Actual Versus Fitted Values for TestTomcat1	46
Figure 28.	F/A-18C Mission Profile (Combat Radius Variable)	47
Figure 29.	F/A-18C Mission Profile (On-Station Endurance Variable)	47
Figure 30.	Actual Versus Fitted Values for TestHornet2	49
Figure 31.	Actual Versus Fitted Values for TestHornet4	51
Figure 32.	SH60B Seahawk Mission Profile (On-Station Endurance Variable)	52
Figure 33.	HH60H Seahawk Mission Profile (Radius of Action Variable).....	52
Figure 34.	SH-60B Weapon Stations and Configurations for TestSeahawk1 and TestSeahawk2	54
Figure 35.	SH-60B Fuel Consumption Rate Versus On-Station Endurance.....	55
Figure 36.	SH-60B Fuel Consumption Rate Versus Payload	56
Figure 37.	Fuel Consumption Versus On-Station Endurance	57
Figure 38.	TestSeahawk2 Actual Versus Fitted Values.....	58
Figure 39.	TestSeahawk3 Fuel Consumption Rate Versus Radius of Action	60
Figure 40.	TestSeahawk3 Fuel Consumption Rate Versus Internal Payload	60

Figure 41.	TestSeahawk4 Fuel Consumption Rate Versus Radius of Action	61
Figure 42.	TestSeahawk4 Actual Versus Fitted Values.....	62
Figure 43.	Effect of Additional Centerline Tank on F/A-18C Combat Radius	64

LIST OF TABLES

Table 1.	Aircraft Java Classes	21
Table 2.	Method Summary for Aircraft	22
Table 3.	TestCobra1 Output	36
Table 4.	Regression Output for TestCobra1	37
Table 5.	Regression Output for TestApache1.java	40
Table 6.	Regression Output for TestSeaStallion2	43
Table 7.	F-14A Weapons Configurations Used in TestTomcat1	45
Table 8.	Regression Output for TestTomcat1	46
Table 9.	F/A-18C Weapon Configurations for TestHornet2 and TestHornet4	48
Table 10.	Regression Output for TestHornet2	49
Table 11.	Regression Output for TestHornet4	51
Table 12.	Regression for TestSeahawk1	55
Table 13.	TestSeahawk2 Regression Output with Indicator Variable.....	58
Table 14.	TestSeahawk4 Regression Output with Indicator Variable.....	62

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

ACAT	Acquisition Category
ACMC	Assistant Commandant of the Marine Corps
AMS	Acquisition Management System
APB	Acquisition Program Baseline
ASD (C3I)	Assistant Secretary of Defense Command, Control, Communication and Intelligence
ASD (S&TR)	Assistant Secretary of Defense Strategy & Threat Reduction
CAE	Component Acquisition Executive
CJCS	Chairman of the Joint Chiefs of Staff
DAB	Defense Advisory Board
DepSecDef	Deputy Secretary of Defense
DoD	Department of Defense
DRB	Defense Review Board
DSB	Defense Science Board
ICAO	International Civil Aviation Organization
J-8	Force Structure, Resources & Assessment Directorate (Joint Staff)
JP-5	(NATO designation: F-44) Jet Fuel used for U.S. Navy aircraft
JROC	Joint Requirements Oversight Council
KTAS	Knots True Air Speed
KPP	Key Performance Parameter
M	Mach Number
MCP	Maximum Continuous Power
MDA	Milestone Decision Authority
ME	Maximum Endurance
MNS	Mission Needs Statement
MOE	Measure Of Effectiveness
MR	Maximum Range
MSL	Mean Sea Level
NATOPS	Naval Air Training and Operating and Procedures Standardization
NAVAIR	Naval Air Systems Command
NWC	Naval War College
NWP	Naval Warfare Publication
OIPT	Overarching Integrated Product Team

ORD	Operational Requirements Document
OT&E	Operational Test & Evaluation
PEO	Program Executive Officer
PM	Program Manager
PPBS	Planning Programming and Budgeting System
R&D	Research & Development
RDT&E	Research, Development, Training & Evaluation
RGS	Requirements Generation System
SAC	Standard Aircraft Characteristics
SECDEF	Secretary of Defense
SR	Specific Range
TAS	True Air Speed
TMS	Type / Model / Series
USD (AT&L)	Undersecretary of Defense, Acquisition, Technology & Logistics
USD (Comptroller)	Undersecretary of Defense, Comptroller
USD (P&R)	Undersecretary of Defense, Personnel & Readiness
USD (Policy)	Undersecretary of Defense, Policy
V	True airspeed (measure in Knots True Air Speed – KTAS)
VCJCS	Vice Chairman, Joint Chiefs of Staff
VCNO	Vice Chief of Naval Operations
VCofS	Vice Chief of Staff
VERTREP	Vertical Replenishment

ACKNOWLEDGMENTS

I would like to extend my sincere appreciation and gratitude to the following individuals, without whom this thesis would not have been possible:

- To my advisor, Professor Lyn R. Whitaker, for her patient guidance and unwavering support. Her vast expertise in regression proved invaluable in my own understanding of data analysis and the development of sound results.
- To Professor Paul J. Sanchez, for his review and supervision of the development of a quality Java model whose methodology could have greater application beyond this thesis.
- To Mr. Jack Keane and Mr. William Kroshl of the Johns Hopkins University Applied Physics Laboratory, for their sponsorship and for providing the means for greater research and the ability to study a “real world” problem.
- To Mr. Tom Rudowski and Mr Michael Caddy of the Naval Air Systems Command, for their critical and timely advice regarding fuel consumption of military aircraft and for providing important research background used in this thesis.
- A special thanks to Dr. Pete Pandolfini of the Johns Hopkins University Applied Physics Laboratory, for generously offering countless hours of his time explaining the finer points of aerodynamic drag to a simple-minded helicopter pilot.
- Finally, to my wife, Patricia, and daughter, Natalie, whose continuous encouragement and support provided me with the confidence I needed to complete this work.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

In January 2001, the Defense Science Board completed their study entitled “More Capable Warfighting Through Reduced Fuel Burden” and made several recommendations for ways of improving fuel efficiency within the Department of Defense (DoD). One of their recommendations was that fuel efficiency should be included in the Requirements Generation Process and that it should be specifically included as a Key Performance Parameter (KPP) on Operational Requirements Documents (ORD) for all future DoD acquisitions.

The Chairman of the Joint Chiefs of Staff (CJCS) has mandated that KPPs should be kept to a maximum of no more than eight for all Operational Requirements Documents. Military aircraft acquisitions are Acquisition Category I (ACAT I) and involve multi-faceted capabilities. Key Performance Parameters for such capabilities must curtail duplication of measure in order to hold the total number of KPPs for a given acquisition program to a minimum.

This thesis develops a Java model utilizing an interface (**Aircraft**) to examine seven current military aircraft flying different mission profiles, each with different weapons payload configurations. Parameters such as combat radius, on-station endurance, and weapons payload are measured for each configuration and mission profile.

Mission radius and on-station endurance are frequently included as KPPs on military aircraft acquisitions. Strong relationships exhibited through regression analysis of these parameters versus fuel efficiency (measured as a consumption rate) demonstrates that fuel efficiency is already measured. Therefore to include fuel efficiency as a KPP for an individual acquisition is a costly duplication of effort.

Furthermore, the methodology utilized in the Java model developed in this thesis could be applied to the development of logistics planning factors for current and future weapons platforms.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

The United States consumes approximately 25% of the world's annual petroleum supply – which is more than the next five highest annual oil-consuming nations combined. Within the United States, the Department of Defense (DoD) is the single largest user of petroleum. Much of this petroleum is consumed as aviation fuel for military aircraft. This thesis addresses the development of a measure of effectiveness for fuel efficiency and the necessity to include fuel efficiency as a Key Performance Parameter for military aircraft acquisition programs.

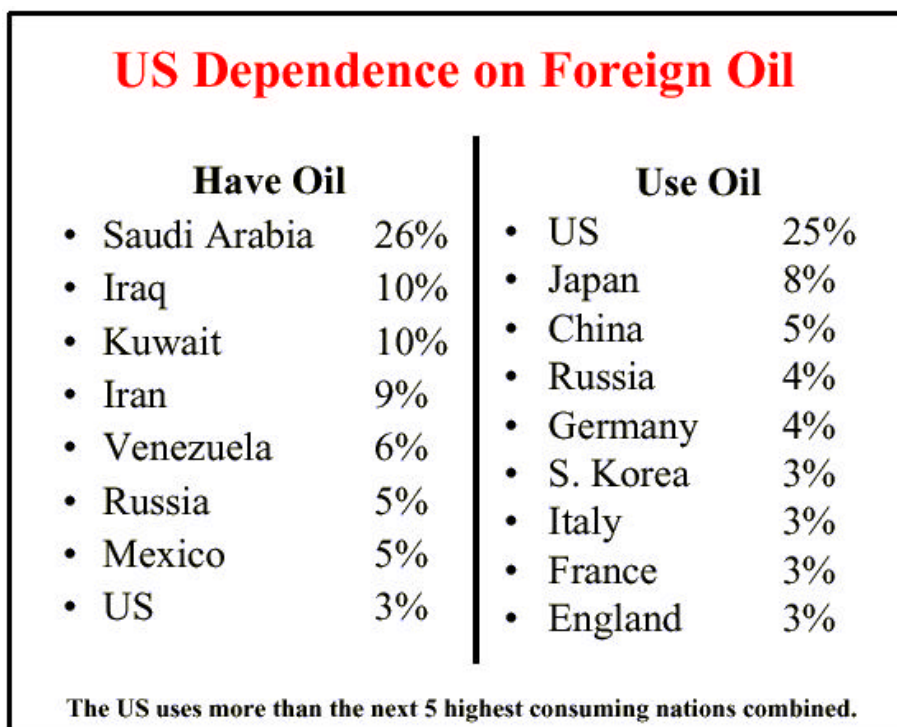


Figure 1. U.S. Dependence on Foreign Oil (From: DSB 2001)

In January 2001, the Defense Science Board (DSB) Task Force on Improving Fuel Efficiency of Weapons Platforms identified significant problems in accounting for fuel consumption within DoD. In their report entitled “More Capable Warfighting Through Reduced Fuel Burden” [DSB 2001] they developed a number of recommendations focusing on the improvement of military capabilities through greater fuel efficiency.

One of their recommendations stated that DoD should explicitly include fuel efficiency in requirements and acquisition processes. Specifically, the DSB recommended that fuel efficiency be “translated into quantifiable and measurable performance criteria and inserted into the requirements determination process. Operational Requirements Documents (ORDs) must directly address efficiency issues at the platform and force levels. The Task Force recommends that DoD develop and apply an efficiency metric for platforms and systems, preferably as a Key Performance Parameter” (KPP) [DSB 2001].

On February 15, 2001, shortly after the DSB Task Force made its recommendations, the Principal Deputy Undersecretary of Defense for Acquisition, Technology and Logistics directed that fuel efficiency would be included as a Key Performance Parameter on all future Operational Requirements Documents [Oliver 2001]. This thesis examines the measurement of fuel efficiency for military aircraft and relationships with frequently used Key Performance Parameters such as combat radius, weapons payload, and on-station endurance. These parameters are shown to have strong correlation with fuel efficiency, and hence any measure of fuel efficiency as a KPP is a costly duplication of effort.

B. BACKGROUND

The Department of Defense (DoD) uses the Acquisition Management System (AMS) to manage all military acquisition programs within the defense resource allocation process. The AMS utilizes a logical structure to develop a capability (weapon system or equipment) from an identifiable requirement. This formal acquisition process utilizes the Requirements Generation System (RGS) in conjunction with the AMS to develop all weapons systems and equipment currently in use by DoD.

When a mission need is identified through a Mission Need Statement, it goes through a very formal validation and approval process within DoD. Once approved, the need (or idea) becomes part of the AMS as an acquisition program (see Figure 1-2). The AMS is composed of three phases called “milestones” – Milestone A, Milestone B, and Milestone C. At each milestone, a gatekeeper – or Milestone Decision Authority (MDA) evaluates the program and determines measures to implement the requirements of the next milestone. As each milestone is reached, additional funding is allocated for the

program. There are different types of acquisitions – based on dollar value of the respective program and these are outlined in Chapter II of this thesis.

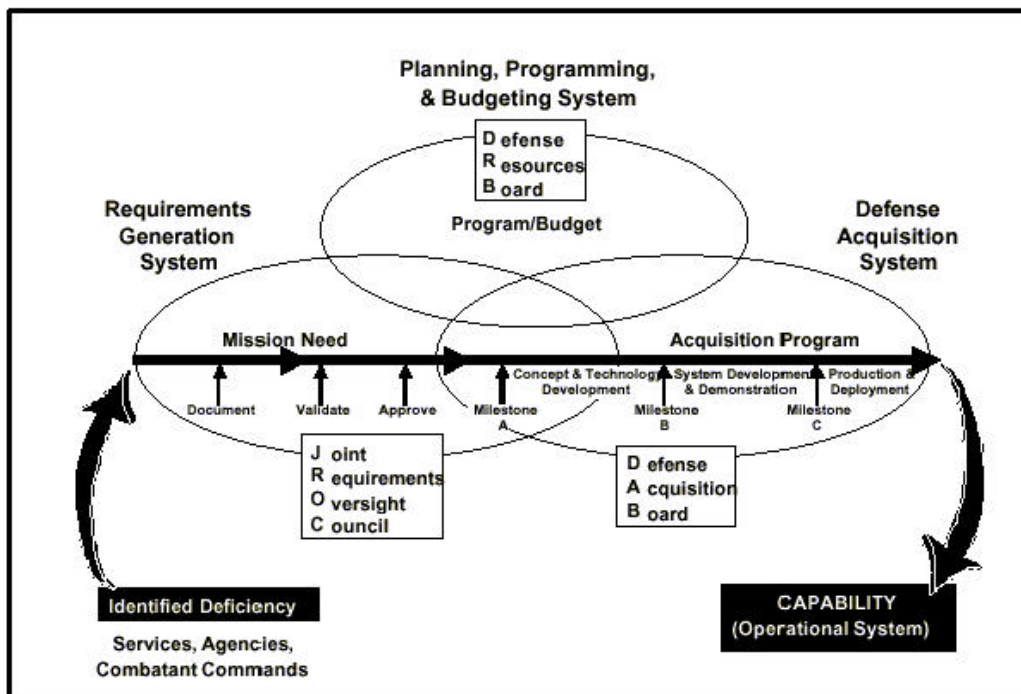


Figure 2. How an Idea Becomes a Capability (From: NWC 2002)

In addition to the MNS, another important document essential to the AMS is the Operational Requirements Document (ORD). The ORD is a “living” manuscript that is evaluated at each milestone of an acquisition program. The ORD translates the broad operational capability from the MNS to specific performance requirements. For example, the ORD for a typical tactical aircraft acquisition program will list specific performance requirements such as: logistics footprint, combat radius, payload, on-station time, etc.

The specific performance requirements listed within the ORD include Performance Parameters and Key Performance Parameters (KPPs). Regarding these two parameters, KPPs are of critical importance. KPPs are those performance parameters established in the ORD considered most essential for successful mission accomplishment by the warfighter. The KPPs are described in terms of threshold and objective values. Failure to meet a KPP threshold can cause the MDA to at best delay the program for reevaluation and at worst to terminate the program. Delays in any defense acquisition program often equate to substantial increases in total life-cycle costs and should be

avoided wherever possible. Furthermore, termination of an acquisition program results in unrecoverable sunk costs. Depending on which stage of the life cycle the program is terminated, these costs can be quite significant. Therefore, it is of utmost importance to consider all aspects of each and every KPP and how they are measured. Important considerations for selecting Key Performance Parameters include:

- Is it essential for defining the system or required capabilities?
- Is it oriented to the warfighter?
- Can it be measured?
- Is it achievable?
- Can it be explained by analysis?
- If the threshold level is not met, is the MDA prepared to cancel the program? [NWC 4-18]

Before fuel efficiency should be included as a KPP, the above questions must be rejoined. This thesis will address this issue with respect to military aircraft acquisitions. Within the field of aviation, the most common measure of fuel efficiency is consumption rate. There are two ways in which fuel consumption rate is measured on aircraft performance charts. Defined as a fuel consumption rate, fuel efficiency can be measured in pounds of fuel consumed per flight hour (lbs/hr). This measure is frequently used when one is interested in determining the amount of fuel consumed in-flight regardless of the distance covered over ground. This measure is commonly used on performance charts for rotary-wing aircraft. Another fuel consumption rate commonly used on fixed-wing aircraft performance charts is specific range. Specific range (SR) is defined as distance units per unit of fuel consumed, and is usually measured in nautical miles per pound of fuel (nm/lb). This measure is frequently used when one is interested in measuring the distance covered over ground (vice the length of flight time) per unit mass of fuel. Each of these measures is convertible to the other and each illustrates distinct characteristics of the aircraft.

The decision to use fuel consumption rate as the Measure of Effectiveness (MOE) for fuel efficiency of military aircraft acquisition programs meets the KPP definition. Fuel consumption rates are quantitative and hence comparison between competing airframes would be possible. Data collected during the Research and Development

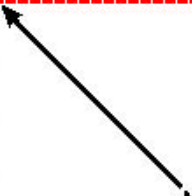
(R&D) phase of an acquisition program would allow for measurable performance of each airframe considered.

C. OBJECTIVE STATEMENT

This thesis will evaluate the KPPs used in a typical Operational Requirements Document for an aircraft acquisition program and determine if fuel efficiency is already measured by these KPPs. By examining different measures of fuel efficiency (fuel consumption rates and specific ranges), the study scrutinizes relationships between these measures of fuel efficiency and factors such as combat radius, payload and on-station loiter time. If there are strong relationships between these factors and fuel efficiency, then any measure of fuel efficiency in the acquisition process for aircraft is quite possibly not a new prospect. Therefore, the introduction of fuel efficiency as a KPP – while well intentioned – is a costly duplication of effort.

As the military is faced with continued operating and procurement budget reductions, the current acquisition methods can quickly become almost cost-prohibitive when an acquisition program is slowed or even cancelled. The ORD should contain the minimum number of KPPs to adequately describe the required capability. To require that all acquisition programs include a KPP for fuel efficiency implies a lack of understanding of the true problem. The Chairman of the Joint Chiefs of Staff has recommended in his instruction, CJCSI 3170.01B that the number of KPPs for an acquisition program should not exceed eight.

Figure 3 contains a listing of the KPPs from recent ACAT1 aircraft acquisition programs. Mission radius – which consists of a combat radius for a specified payload – is a KPP in each case.

No.	RQ-4A Global Hawk	V-22	JSF	F/A-18E/F	F-22A
1	Mission Radius / Endurance	Mission Radius	Mission Radius	Mission 1 Radius	Mission Radius
2	Airspace Coordination	V/STOL Capability	Radar Cross Section	Mission 2 Radius	Radar Cross Section
3	Ground Sta Dyn Control	Cruise Speed	STOVL Performance	Combat Ceiling	Supercruise Mach No.
4	Interoperability	Payload Capacity	CV Suitability	Deck Spot Factor	Sustained Maneuver
5		Self Deployment	Logistics Footprint	Carrier Suitability	Acceleration Time
6		Shipboard Compatibility	Sortie Generation Rate	Recovery Payload	Radar Detection Range
7		Aerial refueling	Mission Reliability	Instantaneous Maneuver	Weapon Loadout
8		Survivability	Interoperability	Specific Excess Power	Logistics Footprint
9		Operational Environment		Acceleration Time	Sortie Generation Rate
10		Precision Navigation		Internal Fuel Capacity	Maintenance per Flight Hr
11		Weapon Sys Reliability		Interoperability	

Note: Per CJCSI 3170.01B (15 April 01), ORD should contain 8 or less KPPs

Figure 3. KPPs from Recent ACAT1 Aviation Programs (From: JHU APL 2002)

This study consists of two parts. In the first part, a Java-based model of various military aircraft currently in inventory will replicate these aircraft flying notional mission profiles, varying the factors of combat radius, payload and on-station loiter time. From each mission, the fuel consumption rate and specific range will be computed. The second part of the study will consist of regression analysis of fuel efficiency (measured as a consumption rate) plotted against the factors combat radius, payload, and on-station time to determine correlation.

For military aircraft, if fuel efficiency is already measured by other KPPs, then to include an additional KPP for fuel efficiency is redundant. This thesis will demonstrate through regression analysis that fuel efficiency (quantified as a fuel consumption rate) is in fact already measured by all of the aforementioned parameters.

D. OVERVIEW

The objective of this thesis is to examine current KPPs used in Operational Requirements Documents for military aircraft acquisition programs and determine the existence of any relationship with fuel efficiency. Chapter II contains an introduction of

DoD acquisition organization with emphasis on the Acquisition Management System (AMS).

In Chapter III, initial research involves analysis of Standard Aircraft Characteristics (SAC) charts. SAC charts are generated for every military aircraft during initial aircraft development and flight-testing. SAC charts contain pertinent data such as power plant specifications, mission descriptions, ordnance capabilities, aircraft basic weight, fuel capacities etc. Among the information contained within each SAC chart are various mission profile specifications for which the aircraft is designed. These mission profiles contain information on the ordnance payload carried, the fuel consumed, total flight time, combat radius (or radius of action), takeoff weight, maximum speed at altitude, etc. The analysis of the SAC charts in Section A of Chapter III examines each mission profile and determines factors such as fuel consumption rate (lbs/hr), combat radius and payload in each case.

Analysis of SAC charts alone is not sufficient to judge variation between mission profiles because of the relatively small number of mission profiles contained in each SAC chart for each aircraft, and because parameters such as fuel consumed for each mission profile for a given aircraft varies significantly between mission profiles. Thus, after exhaustive research of numerous SAC charts, it was determined that a model was needed which could run numerous configurations of the same aircraft through the same mission profile. With this greater number of observations (many different configurations of the same aircraft flying the same mission profile), analysis of the results are substantially enhanced. Chapter III Section B describes in detail a Java-based model developed to allow various military aircraft flying notional mission profiles to be examined.

The results applying the Java model of Chapter III to classes of aircraft in several configurations are analyzed in Chapter IV. The primary tool in this chapter is regression analysis of the factors of fuel consumption rates versus independent variables combat radius, payload, and on-station endurance. Finally, conclusions and recommendations are contained in Chapter V.

THIS PAGE INTENTIONALLY LEFT BLANK

II. THE ACQUISITION PROCESS

In order to fully appreciate the importance of a KPP within an acquisition program, it is essential to understand the function of the DoD acquisition process. The DoD acquisition process is multifaceted with many interacting components, but overall has a very coherent structure. There are two different methods by which the acquisition process commences. The first is through the identification of a requirement, which is then developed into a military capability (weapon system or equipment) to fulfill that requirement. The second method is through utilization of current technology that has potential for future military application. [NWC 4-1]

This chapter will address those characteristics of the DoD acquisition process that are pertinent to understanding the motivation for this thesis. DoD Directive 5000.1 states that the primary purpose of the defense acquisition system is to acquire quality products that satisfy the needs of the operational user with measurable improvements to mission accomplishment, in a timely manner, at a fair and reasonable price. [DoD 5000.1] This process begins with the identification of a need.

Once the need for a specific capability is identified, it is documented in a Mission Need Statement (MNS). The MNS is essentially a very broad description of the required capability. The MNS proceeds through a formal validation process and then enters the AMS prior to reaching Milestone A. The MDA decides whether or not the required capability should be implemented as an acquisition program and advance through the AMS. As the program advances, activities occur as part of the acquisition process to prepare it for the next milestone. The MDA has most of the control of an acquisition program and the Acquisition Category (ACAT) of the program in question determines the designation of the MDA. [NWC 4-5]

A. ACQUISITION PROGRAMS

1. ACAT I Programs

ACAT I programs (Acquisition CATegory I) are the largest and usually most complex acquisition programs that warrant the highest level of oversight. ACAT I programs are designated such when the cost of Research, Development, Test and

Evaluation (RDT&E) is estimated to exceed \$355 million, or the estimated procurement exceeds \$2.135 billion. The Under Secretary of Defense for Acquisition, Technology and Logistics (USD (AT&L)) will designate the MDA for ACAT I programs as one of two choices: 1) himself – USD (AT&L) or 2) the respective DoD component head (service secretary or agency head). If the USD (AT&L) is the MDA, the program is designated ACAT ID. If the DoD component head is designated as the MDA, the program is designated ACAT IC. [NWC 4-5] All major military aircraft acquisition programs are ACAT I.

2. Other Acquisition Programs

Other acquisition programs include: ACAT IA, ACAT II, and ACAT III. ACAT IA is used to designate acquisition programs that are part of the Major Automated Information System (MAIS). These programs have estimated costs in any single year in excess of \$30 million, the total program cost is estimated to exceed \$120 million, or the total life-cycle cost is estimated to exceed \$360 million. [NWC 4-6]

Acquisition programs that do not meet the requirement for ACAT I or ACAT IA are deferred to their respective DoD Component Head (e.g. Secretary of the Navy). The Component Head may so designate such programs as ACAT II if he/she feels that service-level oversight is warranted. An ACAT II program has RDT&E costs estimates exceeding \$135 million or estimated procurement costs in excess of \$640 million. The MDA for ACAT II programs is the Component Acquisition Executive. [NWC 4-6]

ACAT III programs are programs that do not meet the criteria for ACAT I, ACAT IA, or ACAT II. The Component Acquisition Executive designates the MDA to the lowest appropriate level. [NWC 4-6]

B. ACQUISITION RESPONSIBILITIES

Within DoD acquisition programs, there are several responsibilities delegated to various persons and organizational boards. The Secretary of Defense (SECDEF) has overall responsibility and authority for all acquisition programs within DoD. To assist him, the following persons and organizational boards were created: (Note: this list is not all-inclusive and only discusses acquisition organizational structure pertinent to this thesis).

	Defense Resources Board	Defense Acquisition Board	Joint Requirements Oversight Council
Chairman	DepSecDef	USD (AT&L)	VCJCS
Vice Chairman	CJCS	VCJCS	
Executive Secretary	Director, PA&E	Dep. Director, ASM	Director, J-8
Members (Others may participate at the request of the chairman of each respective board.)	Service Secretaries VCJCS USD (AT&L) USD (Policy) USD (Comptroller) USD (P&R)	USD (Comptroller) ASD (S&TR) ASD (C3I) Director, OT&E CAEs (Navy, Army, & Air Force) OIPT Leader PEO PM	VCNO ACMC VCofS Army VCofS Air Force

ACMC: Assistant Commandant of the Marine Corps ASD: Assistant Secretary of Defense ASM: Acquisition Systems Management AT&L: Acquisition, Technology, & Logistics CAE: Component Acquisition Executive CJCS: Chairman of the Joint Chiefs of Staff C3I: Command, Control, Communications, & Intelligence DAB: Defense Acquisition Board DRB: Defense Resources Board J-8: Force Structure, Resources, & Assessment Directorate (Joint Staff)	JROC: Joint Requirements Oversight Council OIPT: Overarching Integrated Product Team OT&E: Operational Test & Evaluation PEO: Program Executive Officer PM: Program Manager P&R: Personnel & Readiness S&TR: Strategy & Threat Reduction USD: Under Secretary of Defense VCJCS: Vice Chairman of the Joint Chiefs of Staff VCNO: Vice Chief of Naval Operations VCofS: Vice Chief of Staff
---	---

Figure 4. Acquisition Oversight (From: NWC 2002)

1. Joint Requirements Oversight Council (JROC)

The JROC assists the Chairman of the Joints Chiefs of Staff (CJCS) in assessing joint military requirements to meet the national military strategy. Among its responsibilities, the JROC is the validation and approval authority for Mission Need Statements for potential ACAT I and ACAT IA programs. Once an ACAT I / IA program MNS is approved by the JROC, it is forwarded to the Defense Acquisition Board (DAB). As major acquisition programs proceed through the acquisition process, the JROC continues to monitor them and assure that they meet the needs of the warfighter. [NWC 4-9]

2. Defense Acquisition Board (DAB)

This is the senior DoD acquisition review board, composed of senior civilian DoD officials and chaired by the USD (AT&L). The DAB provides explicit recommendations to the USD (AT&L) regarding respective acquisition programs. The DAB meets whenever an ACAT ID program reaches a milestone in its acquisition process. At that

time, recommended courses of action are determined regarding the program's future development. [NWC 4-10]

3. Defense Resources Board (DRB)

The DRB is chaired by the Deputy Secretary of Defense (DepSecDef) and reviews DoD's resources in order to advise the SECDEF on major resource allocation decisions. The DRB resolves issues on programs that eventually end up in the President's Budget. "A DRB review can severely impact the budgeting of major acquisition systems as each program must compete with all other programs for funding." [NWC 4-10]

4. Deputy Secretary of Defense (DepSecDef)

DepSecDef is chairman of the DRB and as such approves funding for all new acquisition programs. [NWC 4-11]

5. Under Secretary of Defense for Acquisition, Technology and Logistics (USD (AT&L))

USD (AT&L) is the senior acquisition official and is responsible for establishing all acquisition policies and procedures for DoD. He is the Defense Acquisition Executive for all ACAT ID programs, is the chairman of the DAB and member of the DRB. [NWC 4-12]

6. Chairman of the Joint Chiefs of Staff (CJCS)

The CJCS is responsible for assessing military requirements for defense acquisition programs and represents the Combatant Commanders (CINCs) regarding operational needs. [NWC 4-12]

C. ACQUISITION DOCUMENTS

1. Mission Need Statement (MNS)

As discussed previously, the MNS presents a military operational requirement in its simplest form. It describes the problem in broad terms with little specificity – this comes later in the acquisition process. Anyone within DoD can generate a MNS.

2. Operational Requirements Document (ORD)

As described in Chapter I, the ORD is a "living manuscript" which translates the broadly defined need from the MNS into specific performance requirements. The ORD defines operational parameters as both Performance Parameters and Key Performance Parameters (KPPs). "These parameters reflect capabilities such as range, speed,

probability of kill, platform survivability, weight, etc., and are described in terms of objective values (what is desired) and threshold values (what will be accepted). [NWC 4-17]

3. Key Performance Parameters (KPPs)

KPPs are those performance parameters considered most essential to successful mission accomplishment. KPPs for ACAT I and ACAT IA programs are validated by the JROC. The MDA has final approval of all KPPs within an ORD. KPPs become part of the Acquisition Program Baseline (APB). As stated in Chapter I, failure to meet the threshold level of any KPP is grounds for reevaluation or even cancellation of the acquisition program by the MDA. [NWC 4-18]

4. Thresholds and Objectives

KPPs are defined by threshold and objective values as explained above. These describe the operational goals of the program for performance and cost. Prior to Milestone A, the Program Manager (PM) of an acquisition program proposes threshold and objective values for all KPPs, which must be approved by the MDA. However, the PM can refine the threshold and objective values as the program progresses through the AMS. [NWC 4-18]

5. Acquisition Program Baseline (APB)

This document contains the threshold and objective values for the most important cost, schedule, and performance parameters of an acquisition program. The APB is prepared by the PM and submitted to the MDA. At each milestone decision point the APB is reviewed. Any threshold value not met is grounds for a deviation report by the PM and triggers an unscheduled DAB review. [NWC 4-19]

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODEL DESCRIPTION

A. INITIAL SAC CHART ANALYSIS

SAC charts are governed by MIL-P-2203: Military Specification Performance Data Report for Standard Aircraft Characteristics Charts for Piloted Aircraft (21 September 1959). Figure 5 depicts the cover of the SAC chart for the U.S. Navy T-2B Buckeye trainer aircraft.

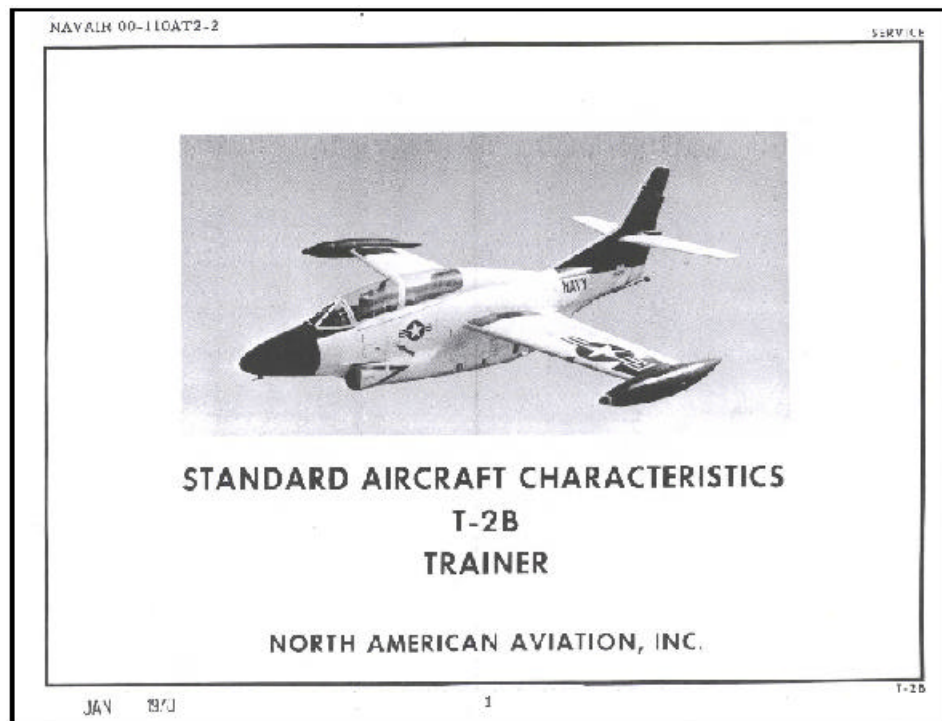


Figure 5. Typical SAC Chart Cover Sheet (From: NAVAIR 1970)

Each SAC chart contains a performance summary in which aircraft parameters are outlined as they pertain to the specified mission profiles. Each mission profile has its own total fuel consumption, payload, total flight time, combat radius, and range data listed in the performance summary. For each mission, fuel consumption rates (measured in either pounds of fuel per flight hour or nautical miles per pound of fuel) can be gleaned from this data. Figure 6 shows an excerpt from the T2B SAC chart with this data highlighted for each of the 4 mission profiles (1-Basic Trainer, 2-Basic Trainer & Rockets, 3-Basic Trainer & Gun Packages, and 4-Basic Trainer & Bombs) outlined within the SAC.

PERFORMANCE SUMMARY					
TAKE-OFF LOADING CONDITION		BASIC TRAINER (Tiptanks)	BASIC TRAINER & ROCKETS	BASIC TRAINER & GUN PACKAGES	BASIC TRAINER & BOMBS
TAKE-OFF WEIGHT	lb.	13,284	13,771	13,771	14,048
Fuel Internal/External	lb.	3166/1326	3166/1326	3166/1326	3166/1326
Payload	lb.	0	252	60	300
Combat range	n.mi.	966 (I)	903 (I)	758 (I)	720 (I)
Average cruising speed	kn.	360	356	352	345
Cruising altitude (avg)	ft.	44,000	43,000	41,500	40,600
Combat radius	n.mi.	498 (II)	337 (III)	235 (IV)	165 (V)
Average cruising speed	kn./kn.	360/360	356/356	352/352	238/263
Mission time	hr.	3.21	2.53	2.19	1.90
LANDING WEIGHT	lb.	9660	10091	10123	10428
Fuel	lb.	788	812	854	872

Figure 6. Excerpt T-2B SAC p. 4 Performance Summary (From: NAVAIR 1970)

The following definitions are useful for interpreting the performance summary of SAC charts [MIL-P-2203]:

- Payload – The load which justifies the mission. Payload includes cargo, personnel other than crew, bombs, rockets, reconnaissance cameras and flares, etc; includes ammunition for fighters and gunnery trainers only; excludes all fuel except that portion carried by a tanker to be transferred.
- Combat Radius – The distance attainable on a practicable flight to the target and return a distance equal to that flown out, carrying a specific load (bombs, cargo, personnel, etc.) to or from the target according to a sequence of operations specified under the mission profile diagrams.
- Combat Range – The distance (including distance covered in climb) attainable on a practicable one-way flight carrying load (bombs, cargo, etc.) the entire distance.
- Total Mission Time – Time in air (excludes time before start of initial climb and reserve fuel unless otherwise specified and noted).

In addition to the performance summary, the SAC charts also contain diagrams of the mission profiles, showing specific flight evolutions required in the execution of each mission. Essentially, the flight evolutions where fuel is consumed are displayed on the mission profile diagrams. All climb, cruise, loiter, and combat evolutions are shown along with their required parameters and detail the fuel used, distance covered over ground and time required to complete the evolution. All cruise, loiter and climb evolutions occur at the altitudes specified on the diagram.

SAC charts are written with the following assumptions [MIL-P-2203]:

- Missions are defined in terms of combat radius and combat range problems.
- All flight mission profiles occur under Standard Atmospheric Conditions defined by the International Civil Aviation Organization (see Appendix A)
- All descents are considered “no credit” – specifically no fuel is used and no distance is gained during a descent evolution (standard engineering practice).
- For loiter and combat evolutions, no distance is credited but fuel is consumed.
- No credit in distance shall be given for any expenditure of ammunition.

SAC charts were collected from Naval Air Systems Command (NAVAIR) for forty-three different military aircraft. For each aircraft Type, Model and Series (TMS) the respective SAC chart lists specific mission profiles that were developed during the initial acquisition and testing of each aircraft. From each mission profile, payload and combat radius are given. Fuel consumption rate (in lbs per flight hour) is computed as follows:

$$BR = \frac{(FWto - FWldg)}{FHtot}$$

where: BR = fuel consumption rate (burn rate)

$FWto$ = fuel weight at takeoff

$FWldg$ = fuel weight at landing

$FHtot$ = total flight hours (duration of flight)

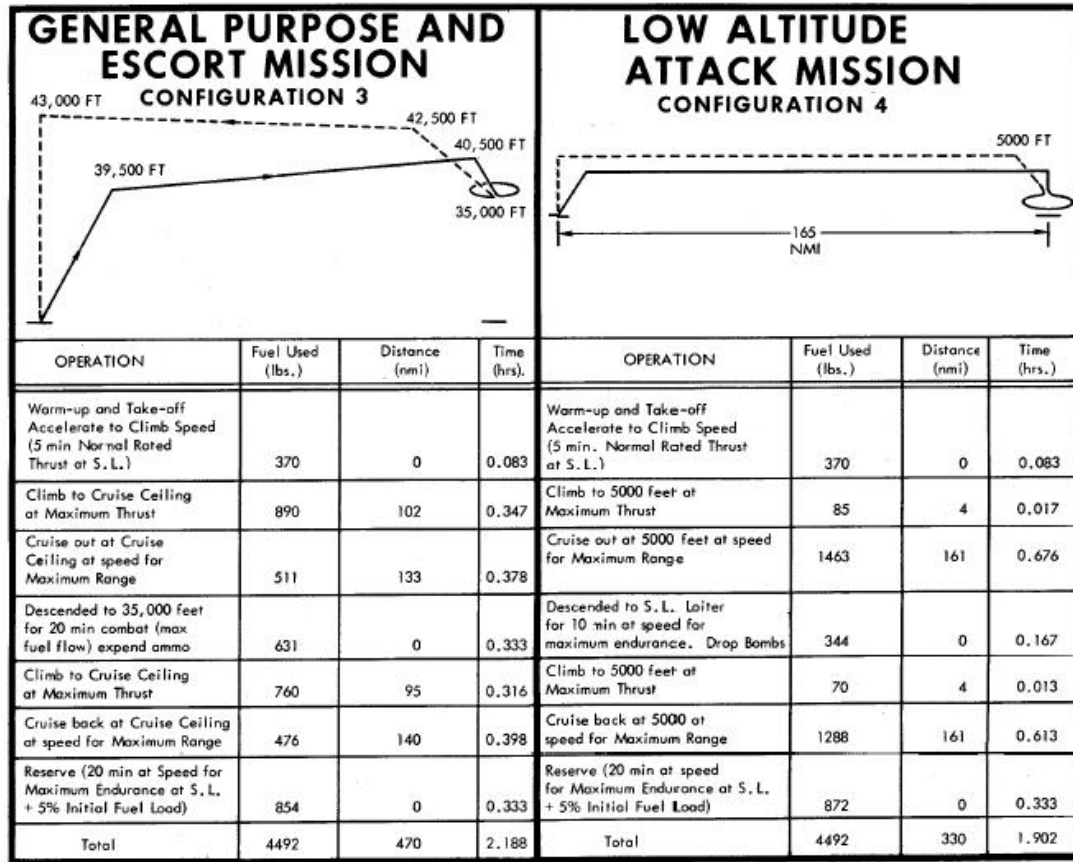


Figure 7. Excerpt T-2B SAC p. 8 Mission Profiles (From: NAVAIR 1970)

Initially, each of the mission profiles for an individual aircraft SAC chart were examined to determine if any relationships exist between fuel consumption rates and the parameters for combat radius and payload. Figures 8 and 9 are plots of burn rate versus combat radius and payload respectively for four different configurations and mission profiles for the F-14D Tomcat and for 6 different configurations and mission profiles for the F/A-18C Hornet. These plots are taken from data for each individual aircraft's SAC chart.

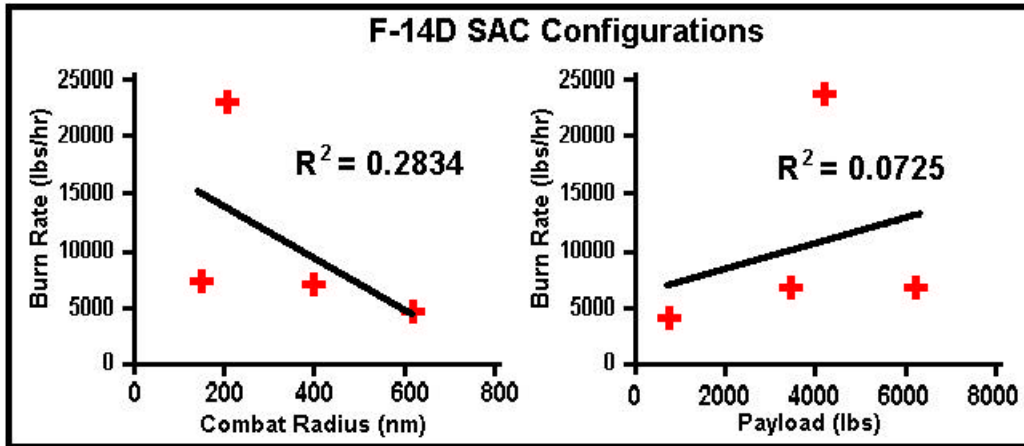


Figure 8. F-14D SAC Chart Burn Rate Versus Combat Radius and Payload

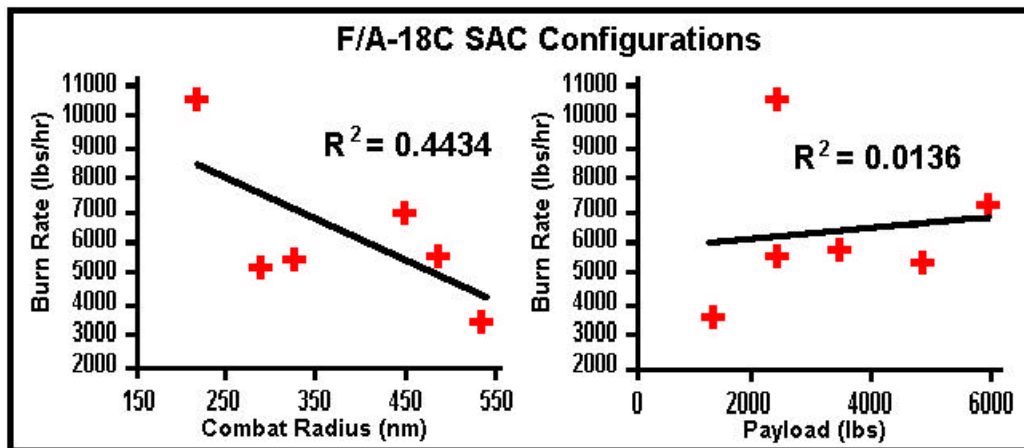


Figure 9. F/A-18C SAC Chart Burn Rate Versus Combat Radius and Payload

Figures 8 and 9 demonstrate the weak relationships exhibited between fuel efficiency and factors such as combat radius and payload – using data taken from the SAC charts. The small R^2 values in addition to the small number of observations for these regression plots illustrate the problem of unexplained variance for this data. Results like this are not uncommon for almost all of the SAC chart information. In fact, the adjusted R^2 values for three of the four plots above are negative. At first glance, examination of these regression plots could lead one to believe that a KPP for fuel efficiency is needed, because it appears to contain information not found in the most closely related factors: combat radius, payload, and on-station endurance.

In almost every case, the SAC chart information for individual aircraft mission profiles demonstrates weak relationships between burn rates and the factors combat

radius and payload. Other ways of examining the SAC Chart data are explored such as aggregating the aircraft together based on similar missions. Using the same procedures outlined for the individual aircraft mission profiles; the burn rates for each aircraft type and mission profiles are graphed versus the factors of Combat Radius and Payload. The results of this analysis are also inconclusive. No discernible relationships are readily apparent. The samples contain too many different aircraft flying too many different profiles to provide any useful information.

However, from the initial analysis completed on individual aircraft SAC charts, there appears to be credible association - albeit slight - between Burn Rate and the factors combat radius and payload. Although the R^2 values for the relationship of Burn Rate versus Combat Radius and Payload in each case are not significantly different than zero, Figures 8 and 9 show general trends observed in all of the SAC Chart data. Specifically, in each case it was observed that fuel efficiency (measured as consumption rate) was negatively related to combat radius and positively related to payload - as one might suspect. A plausible explanation for the small R^2 is that the larger variation of the burn rates is due to the fact that in each case the aircraft are flying different mission profiles and in almost all cases, the total fuel consumed in-flight is not constant across those profiles and configurations.

Further examination of these relationships using a model which can generate a greater number of cases with more control over input parameters such as mission profile, fuel consumed, and payload provides more robust results.

B. JAVA MODEL

Using the respective NATOPS performance graphs for a sample of aircraft Type/Model/Series (TMS) that are currently in the U.S. inventory, a Java-based model was developed that would replicate different configurations of an individual aircraft on a given flight mission profile – similar to profiles found in the SAC charts for respective aircraft. The model consists of a single interface (**Aircraft**) and individual aircraft classes. The aircraft developed for this model experiment are displayed in Table 1.

TMS	Nomenclature	Primary Mission	Java Class
AH-1W	Cobra	USMC Attack Helicopter	AH1W.java
AH-64A	Apache	US Army Attack Helicopter	AH64A.java
CH-53E	Super Stallion	Heavy Lift Helicopter	CH53E.java
F-14A	Tomcat	Strike Fighter	F14AStd.java
F/A-18C	Hornet	Strike Fighter	FA18CStd.java
HH-60H	Seahawk	Special Warfare	HH60H.java
SH-60B	Seahawk	Sea Control & Surveillance	SH60B.java

Table 1. Aircraft Java Classes

Each of the aircraft Java classes implements the interface, **Aircraft**, to accomplish the execution classes – which are individual mission profiles designed along the same methodology used by the SAC charts (in accordance with the assumptions defined by MIL-P-2203). The interface, **Aircraft**, requires the individual aircraft classes to have specific methods. For example, all aircraft classes have a **climb** method, which determines the time elapsed, fuel used, and distance traveled in a climb from its current altitude to a specified altitude. The methods utilized by **Aircraft** are listed in Table 2.

Essentially, the basic methods that are utilized in every mission profile (java execution class) are **climb**, **cruise**, **loiter**, combat (called **speedyTime** within **Aircraft**) and maximum power cruise (called **speedyDist** within **Aircraft**). There are other methods that individual aircraft execute such as hover evolutions (only performed by helicopters) and those methods are contained within their respective aircraft classes. The Javadocs for all of the aircraft classes are detailed in Appendix B.

The key methods used by **Aircraft** utilize the specific aircraft classes to compute the three variables necessary to determine the fuel consumption rate for a given aircraft flying a given mission profile. Those variables are (1) total flight time elapsed, (2) total distance traveled, and (3) total fuel used during the specified mission profile outlined in the execution class. Time, distance and fuel are continuously computed and updated during each flight evolution. The flight evolutions (java methods) occurring in the execution classes in which flight time, distance and fuel-consumed changes are **climb**, **cruise**, **loiter**, **speedyTime** and **speedyDist**.

Method Summary	
double[]	<u>climb</u> (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	<u>cruise</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	<u>descend</u> (double descendTo) This is a void method which simply changes the aircraft altitude.
double	<u>getClimbDist</u> (double alt) Returns the distance traveled over ground during a climb to the specified altitude (alt).
double	<u>getClimbFuel</u> (double alt) Returns the fuel consumed during a climb to the specified altitude (alt).
double	<u>getClimbTime</u> (double alt) Returns the time elapsed during a climb to the specified altitude (alt).
double	<u>getCurrentAlt</u> () Returns altitude (the current altitude of the aircraft).
double	<u>getCurrentGS</u> () Returns groundSpeed (the current speed over ground of the aircraft - equal to True Air Speed).
double	<u>getCurrentGW</u> () Returns grossWt (the current gross weight of the aircraft).
double[]	<u>getLevelFitPerf</u> (double alt) Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed (Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmc _p), and Maximum Continuous Power Fuel Flow (FFmcp).
double	<u>getSurfTemp</u> () Returns surfTemp (surface temperature at takeoff airfield) instance variable.
double	<u>interpolate</u> (double lower, double percent, double upper) Returns the interpolated value between the lower and upper doubles.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

Table 2. Method Summary for **Aircraft**

The key methods used by **Aircraft** utilize the specific aircraft classes to compute the three variables necessary to determine the fuel consumption rate for a given aircraft flying a given mission profile. Those variables are (1) total flight time elapsed, (2) total distance traveled, and (3) total fuel used during the specified mission profile outlined in the execution class. Time, distance and fuel are continuously computed and updated during each flight evolution. The flight evolutions (java methods) occurring in the execution classes in which flight time, distance and fuel-consumed changes are **climb**, **cruise**, **loiter**, **speedyTime** and **speedyDist**.

1. Java Method: climb

The method climb takes an argument which is the assigned altitude for the aircraft object. The method first determines the current aircraft altitude and, using the matrices

within the specific aircraft class, then determines the flight time elapsed, distance covered, and fuel consumed in the climb evolution. The variables (time, distance and fuel) are direct table look-up values based on altitude, aircraft gross weight, outside air temperature (for the helicopters), and drag index (for the fixed wing aircraft). The matrices for these variables were developed from the NATOPS climb performance graphs for each respective aircraft. The climb method returns an array of time elapsed, distance traveled, and fuel consumed during the entire climb evolution.

For example, given an AH-1W with gross weight of 14,000 lbs executing a climb from sea level to 6,000 feet under standard atmospheric conditions, the time elapsed, distance traveled and fuel consumed during the climb is approximately three minutes, four nautical miles and fifty-four pounds of JP-5 jet fuel respectively (see Figure 10).

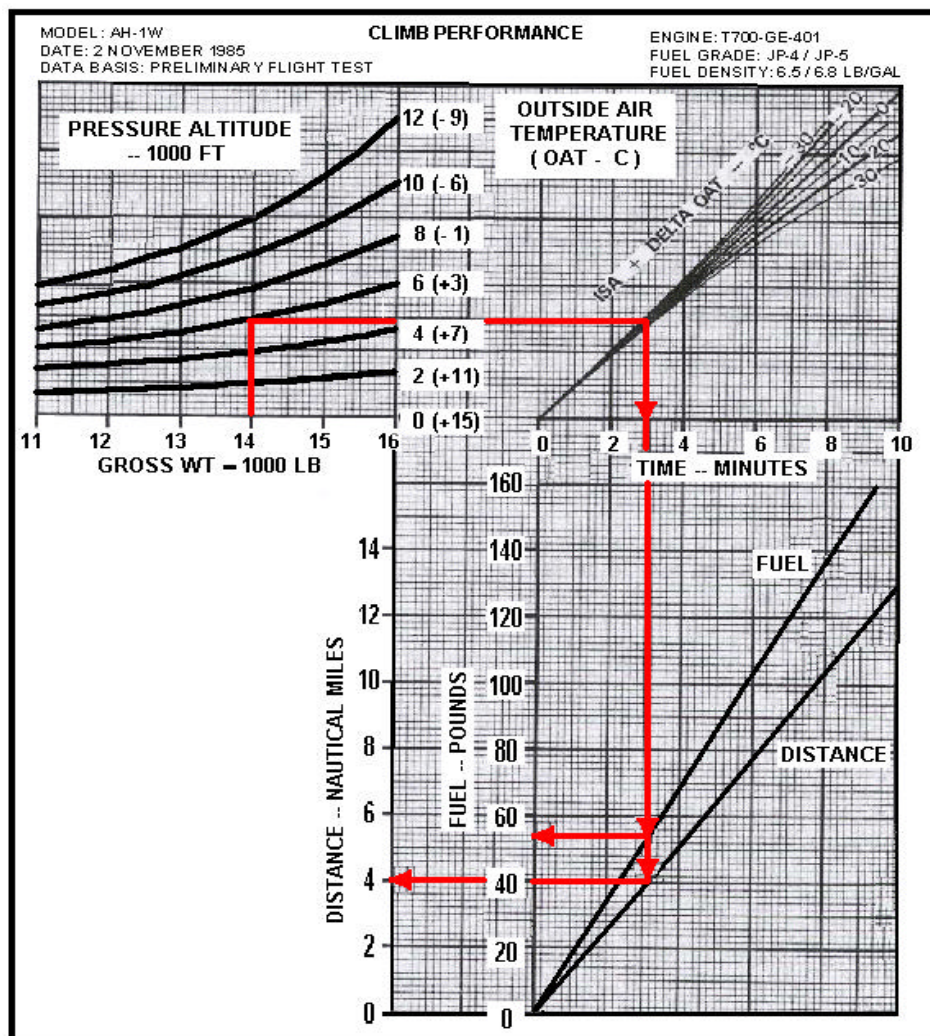


Figure 10. AH-1W Climb Performance (From: NAVAIR 2001)

2. Java Methods: loiter, speedyTime, cruise and speedyDist

The methods **loiter** and **speedyTime** both take elapsed time as the argument for the flight evolution. Both of these flight evolutions occur over a specified amount of time. The **loiter** method is used to model flight performance where on-station time is measured – where the amount of time spent airborne is the objective. The **speedyTime** method is used to model air combat situations and evasive maneuvering where time rather than distance traveled is the crucial element of that segment of the mission profile.

The methods **cruise** and **speedyDist** both take distance traveled as the argument for the respective flight evolution. Both of these flight evolutions occur over a specified distance. The **cruise** method is used to model flight performance during level flight from one geographic point to another. The **speedyDist** method models the same flight performance – except that the power setting is increased to maximum continuous.

The following assumptions apply to these methods:

- Altitude remains constant during these flight evolutions (cruise, loiter, speedyTime and speedyDist)
- All loiter flight evolutions occur at maximum endurance airspeed (V_{ME}) and fuel flow (FF_{ME})
- All cruise flight evolutions occur at maximum range airspeed (V_{MR}) and fuel flow (FF_{MR})
- All speedyTime and speedyDist flight evolutions occur at maximum continuous power airspeed (V_{MCP}) and fuel flow (FF_{MCP})

These last three assumptions presuppose that the aircraft is optimally flown in light of the continuously changing gross weight of the aircraft as a result of the constant fuel consumption. As the gross weight of the aircraft changes, the optimal airspeed for the given conditions (altitude, outside air temperature, drag index, etc.) will also change.

For example, when the mission profile calls for remaining on station, the **loiter** method is employed. During this **loiter** flight evolution; the gross weight of the aircraft is reduced as fuel is consumed. Consequently, the optimum airspeed and fuel flow for maximum endurance conditions changes. The airspeed and fuel flow must be changed accordingly to maintain optimal flight characteristics. The Java model accounts for this by updating the maximum endurance airspeed and fuel flow using a “stepwise”

technique. A “for” loop within the **loiter** method updates the incremental changes in gross weight and re-computes the optimal airspeed and fuel flow using a table look-up technique similar to that used in the **climb** method.

Figure 11 shows a level flight performance chart for the HH-60H Seahawk helicopter. For this example, an HH-60H with a gross weight of 18,000 lbs, flying at 4,000 feet pressure altitude with an outside air temp (OAT) of 30 degrees centigrade will have the following optimal flight parameters:

- Maximum endurance airspeed (V_{ME}) \cong 77 KTAS (Knots True Air Speed)
- Maximum endurance fuel flow (FF_{ME}) \cong 830 lbs/hr (pounds per hour)
- Maximum range airspeed (V_{MR}) \cong 133 KTAS
- Maximum range fuel flow (FF_{MR}) \cong 1,098 lbs/hr
- Maximum continuous power airspeed (V_{MCP}) \cong 143 KTAS
- Maximum continuous power fuel flow (FF_{MCP}) \cong 1,220 lbs/hr

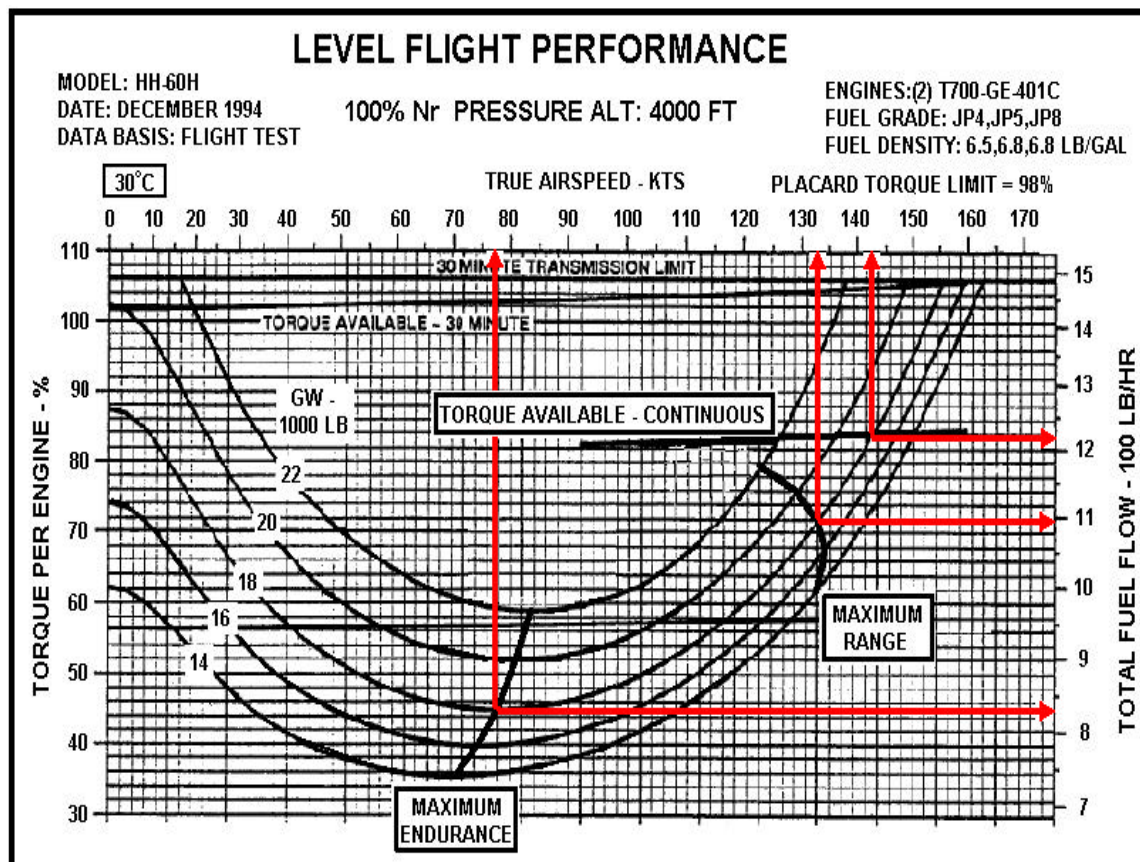


Figure 11. HH-60H Level Flight Performance (NAVAIR 2001)

The graph featured in Figure 11 is one of many such graphs for the HH-60H identifying specific flight parameters (airspeeds and fuel consumption rates) for different altitudes and air temperatures. Each of these graphs is represented by a matrix within the individual aircraft Java class (in this case HH60H). Each matrix has the optimal airspeed and fuel consumption rates classified by gross weight. Interpolation between the gross weight lines provides a close approximation of the actual optimal airspeed and fuel consumption rate. The gross weight is reduced as fuel is consumed – and consequently the optimal airspeeds and fuel consumption rates also change. The **loiter**, **cruise**, **speedyTime** and **speedyDist** methods account for these changes by incrementally updating the optimal airspeed and fuel consumption rate.

For each individual helicopter, separate level flight performance graphs exist based on pressure altitude and outside air temperature. For the fixed-wing aircraft (EG: F/A-18C) these graphs are similar but the parameters displayed are different. Figure 12 shows level flight performance for an F/A-18C Hornet.

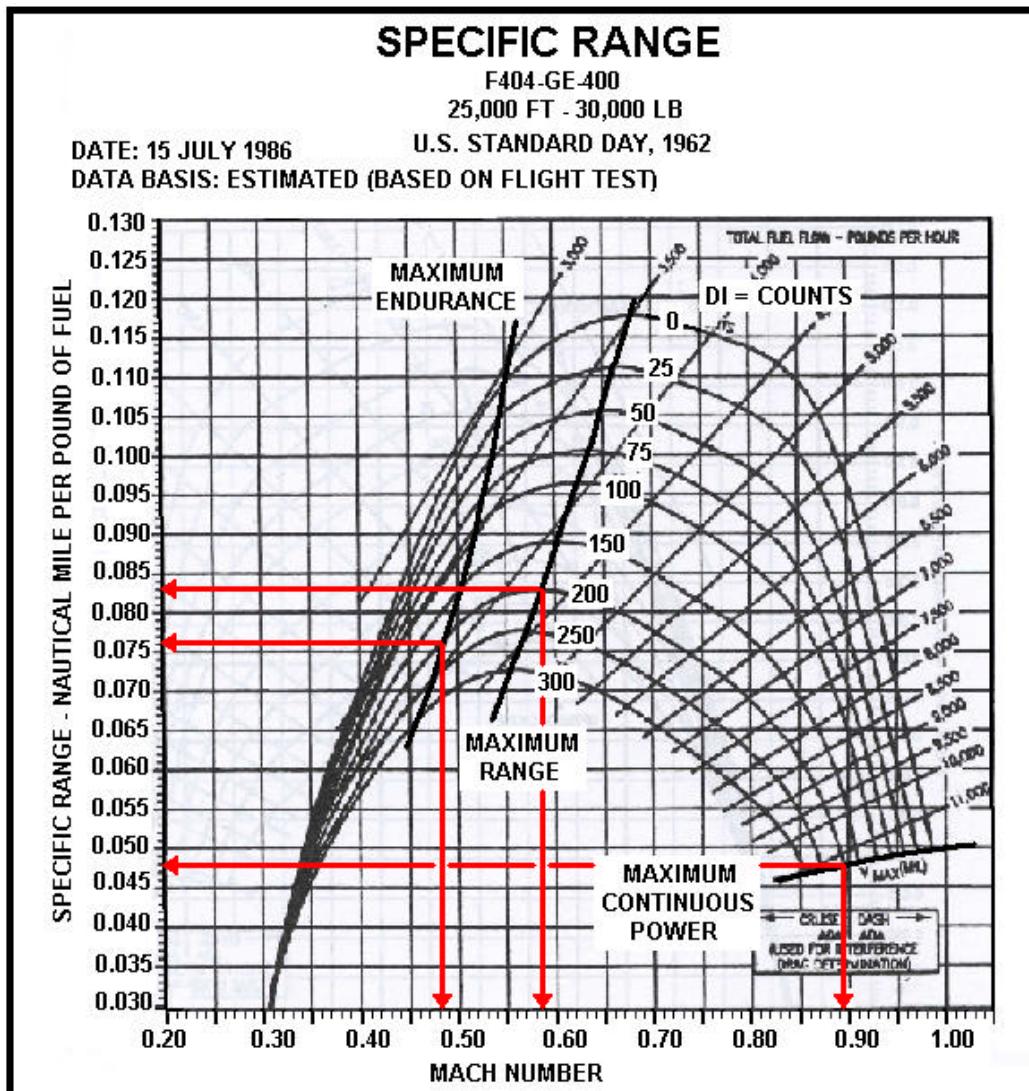


Figure 12. F/A-18C Level Flight Performance (From: NAVAIR 2001)

In the example highlighted on Figure 12, an F/A-18C with gross weight of 30,000 pounds, flying at 25,000 feet in a standard atmosphere with a cumulative drag index of 200 counts will have the following optimal parameters:

- Maximum endurance Mach number (M_{ME}) $\cong 0.482$
- Maximum endurance specific range (SR_{ME}) $\cong 0.0762$ nm/lb (nautical miles per pound of fuel)
- Maximum range Mach number (M_{MR}) $\cong 0.586$
- Maximum range specific range (SR_{MR}) $\cong 0.0830$ nm/lb
- Maximum continuous power Mach number (M_{MCP}) $\cong 0.894$
- Maximum continuous power specific range (SR_{MCP}) $\cong 0.0478$ nm/lb

Mach number is defined as the ratio of aircraft speed to the speed of sound at altitude in a standard atmosphere. Therefore, the conversion of Mach number to true airspeed takes the following form:

$$V = M \times a$$

where: V = true airspeed (in KTAS)

M = Mach number (from the graph)

a = speed of sound at altitude (Standard Day – Appendix A)

Dividing the true airspeed by the specific range yields the fuel flow in pounds per hour:

$$FF = \frac{V}{SR}$$

where: FF = fuel flow (in lbs/hr)

SR = specific range (in nm/lb – from the graph)

Using these conversions for the example in Figure 12:

- Maximum endurance airspeed (V_{ME}) \cong 290 KTAS (Knots True Air Speed)
- Maximum endurance fuel flow (FF_{ME}) \cong 3,808 lbs/hr (pounds per hour)
- Maximum range airspeed (V_{MR}) \cong 353 KTAS
- Maximum range fuel flow (FF_{MR}) \cong 4,250 lbs/hr
- Maximum continuous power airspeed (V_{MCP}) \cong 538 KTAS
- Maximum continuous power fuel flow (FF_{MCP}) \cong 11,255 lbs/hr

Like Figure 11, the graph featured in Figure 12 is one of many such graphs for the F/A-18C identifying specific flight parameters (Mach numbers and specific ranges) for different altitudes and aircraft gross weights. It is important to note that the level flight performance graphs for the F/A-18C and the F-14A both assume standard atmospheric conditions. These conditions are based on an outside air temperature of 15 degrees Celsius at sea level with a change of -1.978 degrees Centigrade per 1000-foot increase in altitude. Appendix A details the properties of the standard atmosphere (also called the “U.S. Standard Day, 1962”).

Each of these graphs is represented by a matrix within the individual aircraft Java class (in this case FA18Cstd). Each matrix has the optimal specific ranges and Mach numbers classified by drag index. Interpolation between the drag index lines provides a

close approximation of the actual optimal Mach number and specific range for a given gross weight and altitude. Interpolation between matrices representing the different gross weights and altitudes returns a close approximation to the aircraft operating at a given altitude and gross weight. The gross weight is reduced as fuel is consumed – and consequently the optimal Mach numbers and specific ranges also change. The **loiter**, **cruise**, **speedyTime** and **speedyDist** methods account for these changes by incrementally updating the optimal Mach number and specific range. Finally, the Mach number and specific range values are converted to airspeed and fuel consumption rate for use in the **cruise**, **loiter**, **speedyTime** and **speedyDist** methods.

3. Aerodynamic Drag

Aerodynamic drag is a major factor influencing fuel consumption rates – particularly for fast-moving aircraft and for aircraft whose cross-sectional drag area varies greatly with different payloads. The flight performance graphs for almost all of the aircraft had some means of accounting for drag. The CH-53E, F-14A, F/A-18C, AH-1W and AH-64A all have dependence on drag developed into their flight performance graphs. The determination of the drag index is based on the payload carried by the aircraft. This section will demonstrate how the drag index is computed for the aircraft used in the Java model.

a. Drag for the SH-60B and HH-60H Seahawk Helicopters

The performance graphs for the SH-60B and HH-60H Seahawk do not account for drag. The relatively slower airspeed of the Seahawk helicopters relative to the fixed wing aircraft combined with the smaller variation in cross-sectional drag area for given payloads equates to trivial changes in the overall drag coefficient for these aircraft. The only situation where drag might have a greater effect on fuel efficiency would occur when the helicopters are utilized for vertical replenishment (VERTREP) over long distances. Neither the HH-60H nor the SH-60B NATOPS manuals list drag performance graphs for VERTREP – as this is not a primary mission area for these aircraft.

b. Drag for the CH-53E Super Stallion Helicopter

The performance graphs for the CH-53E Super Stallion have a Special Mission section where VERTREP flight performance charts are listed. The computation

of the drag is based on the frontal area of the external load and its handling characteristics.

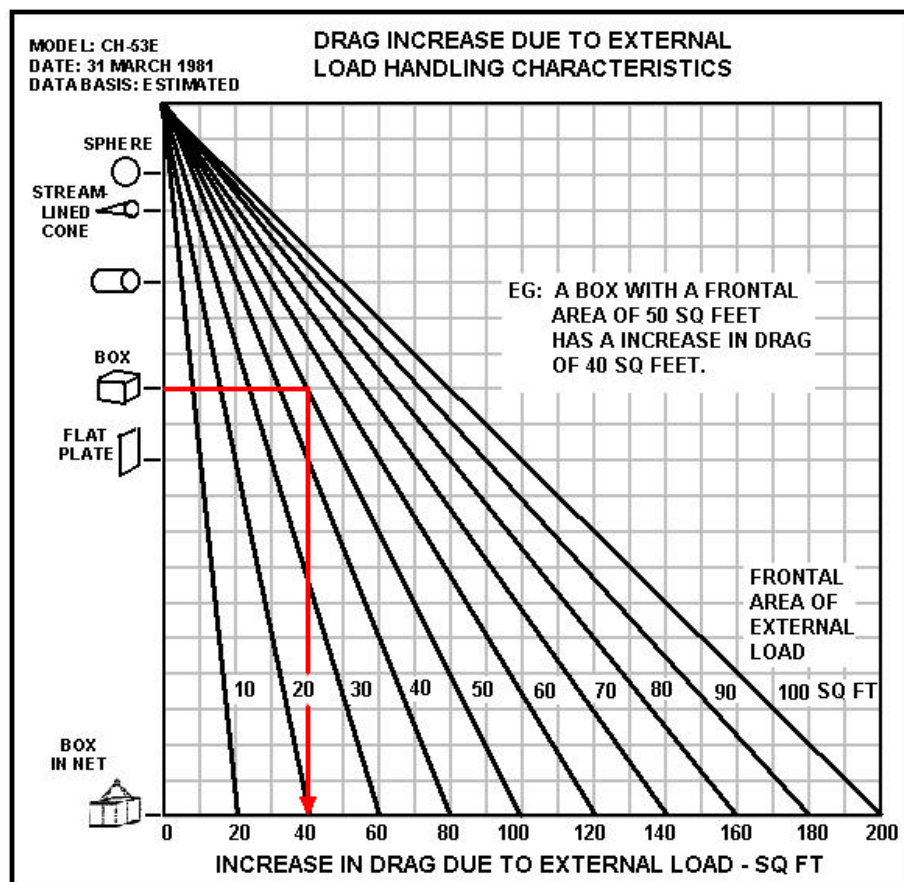


Figure 13. CH-53E External Load Drag (From NAVAIR 2001)

Once the increase in drag is determined, special external load flight performance graphs are based on the total fuel consumption due to the load. The **vertrep** method within CH53E assumes a constant airspeed based on maximum safe speed for external operations (arbitrarily determined by the author). Interpolation between gross weight, altitude and computed drag area is done using the same algorithm – by converting these special flight performance graphs to matrices within CH53E and then interpolating between values for a given gross weight within an individual matrix and between matrices for a given altitude and drag area. Figure 14 is an example of an external load flight performance graph.

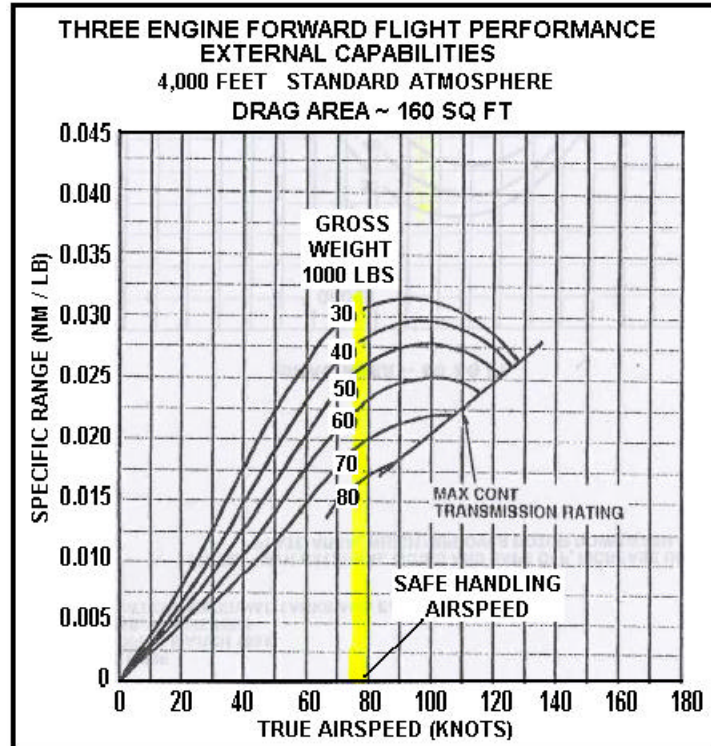


Figure 14. CH53E External Load Flight Performance (From: NAVAIR 2001)

c. Drag for the AH-1W Cobra and AH-64A Apache Helicopters

The AH-1W Cobra and the AH-64A Apache helicopters both account for drag within their respective level flight performance graphs. A “multiplying factor” is determined based on the drag configuration. This multiplying factor is used within the level flight performance graphs to determine the change in power required to maintain required airspeed (such as VMR). This relationship is nicely fitted with a least squares polynomial regression equation. Once the change in power required is determined the performance graph functions exactly like the performance graphs for the other helicopters. V_{ME} , FF_{ME} , V_{MR} , FF_{MR} , V_{MCP} and FF_{MCP} are determined by interpolation of table look-up values converted to matrices in the respective Java classes for each aircraft.

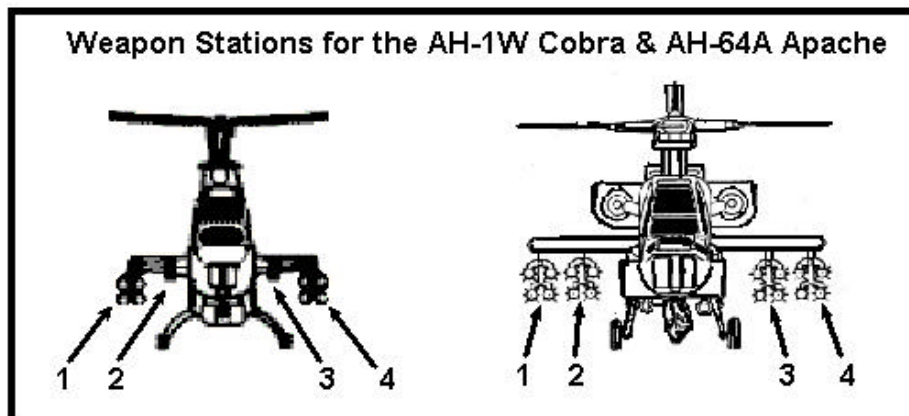


Figure 15. AH-1W and AH-64A Weapons Stations

d. Drag for the F-14A Tomcat and F/A-18C Hornet

The flight performance graphs for both the F-14A Tomcat and F/A-18C Hornet are based on drag index (or “drag count”). For each of these aircraft, the drag count is based on the ordnance loaded on the aircraft. The NATOPS performance charts list all of the ordnance that the aircraft are capable of carrying and the corresponding drag count for each type of ordnance plus the specific weapons pylons and delivery mechanisms used to secure the ordnance to the aircraft fuselage.

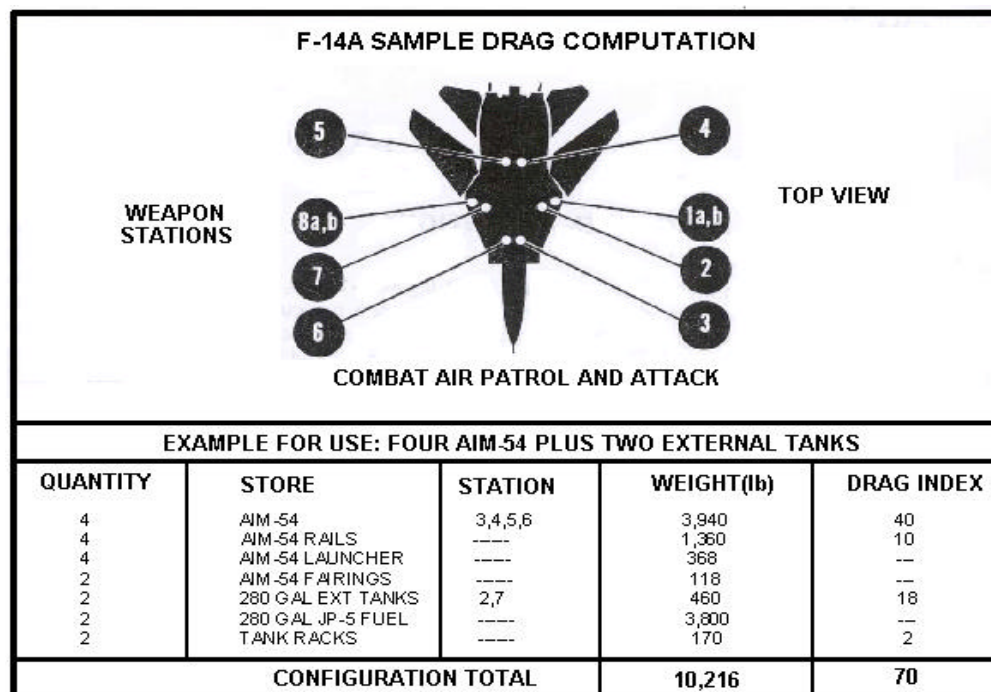


Figure 16. F-14A Sample Drag Computation (From: NAVAIR 2001)

The drag count for the F-14A is easily computed (see Figure 16) and assigned as a parameter to the constructor for **F14Astd** (the Java class for the F-14A Tomcat). However, drag for the F/A-18C is broken down into two separate types – basic stores drag and interference drag. The basic stores drag is computed in the same manner as the total drag for the F-14A – by adding up the basic stores drag counts for each type of ordnance used for a specified weapons load out.

Interference drag only occurs for certain weapons payloads and only at specific weapons stations. The F/A-18C flight performance graphs include a matrix, which contains a listing of all possible combinations of ordnance mixes where interference drag occurs. The interference drag counts are summed to get an interference drag code (see Figure 17). This interference drag code is then used to determine the interference drag, which is dependent upon airspeed. Interference drag is added to basic stores drag to determine the total drag count. Both the Basic Stores Drag Count and the Interference Drag Code are included in the constructor for **FA18Cstd**.

As airspeed changes, interference drag changes, and consequently total drag count changes. FA18Cstd accounts for the changing total drag by re-computing the interference drag incrementally within the **loiter**, **cruise**, **speedyTime**, and **speedyDist** methods.

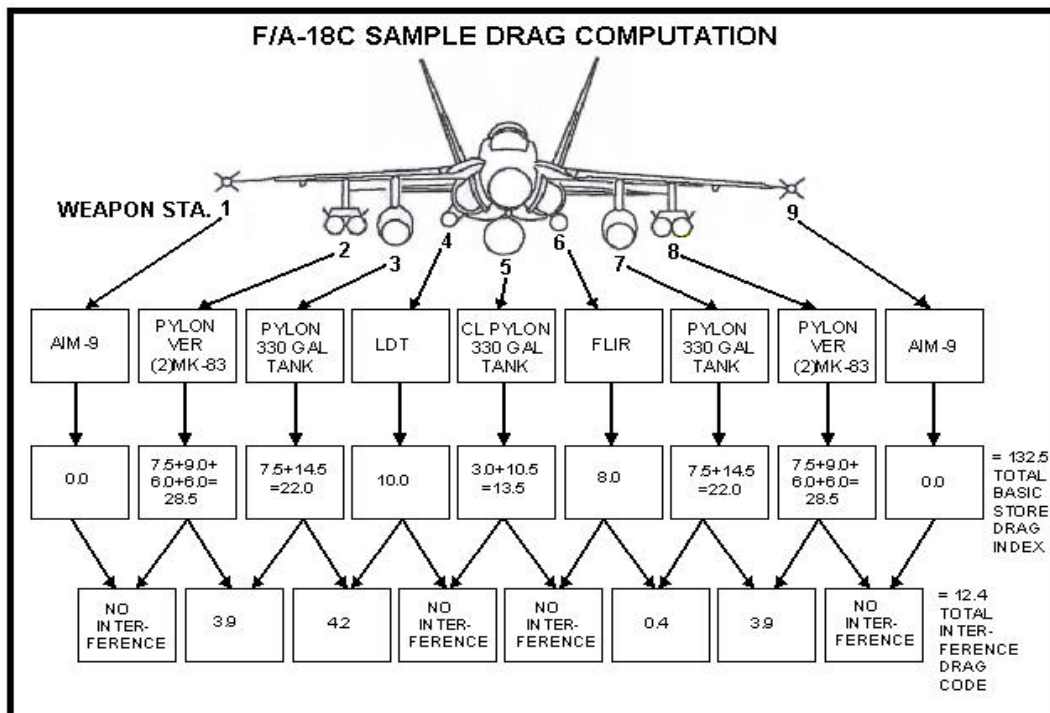


Figure 17. F/A-18C Sample Drag Computation (From: NAVAIR 2001)

IV. MISSION PROFILES AND REGRESSION ANALYSIS

The Java model consisting of the interfaces, **Aircraft** and **Helicopter**, and the individual aircraft classes (**AH1W**, **AH64A**, **CH53E**, **F14Astd**, **FA18Cstd**, **SH60B** and **HH60H**) are used in the creation of Java execution classes. These execution classes represent different mission profiles for each of the aircraft and allow analysis of their respective fuel consumption rates versus the factors of payload, combat radius, and on-station endurance. Each execution class examines several different configurations of each aircraft type flying the same mission profile. The Golden Section Algorithm is used to determine combat radius and on-station endurance for the given profile and drag configuration, holding total fuel consumed to the allotted fuel (within 1.0 lb JP-5). Regression analysis on the results of each model run yields strong correlation of fuel consumption rates (fuel efficiency) to these factors. This chapter outlines the different mission profiles (execution classes) that are used in the evaluation and reports the results of the regression analysis.

A. AH-1W COBRA MISSION PROFILE

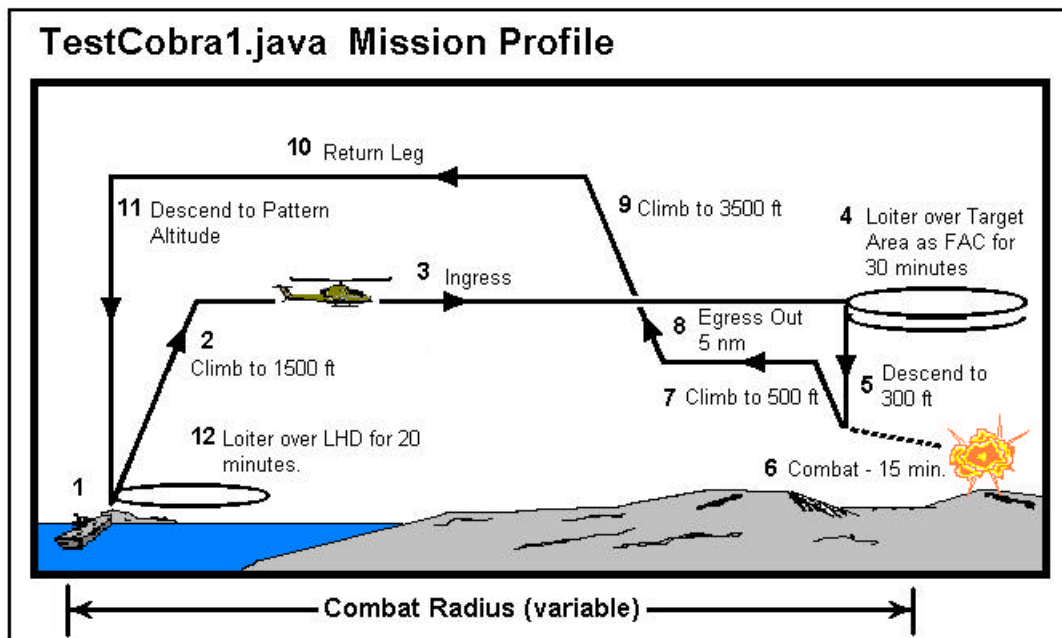


Figure 18. AH-1W Mission Profile

Figure 18 displays a notional mission profile for the AH-1W Cobra helicopter. This mission profile was written into the Java execution class **TestCobra1** and the output

is displayed in Table 3. **TestCobra1** examines sixteen different payload configurations for the AH-1W and determines the maximum possible combat radius (in nautical miles) for each. Fuel used is held constant across all configurations (within 0.5 lbs). The only parameter allowed to vary is combat radius – based on fuel consumption.

TestCobra1 consists of the following flight evolutions:

- Warm up taxi and takeoff
- Climb to 1, 500 ft MSL (Mean Sea Level)
- Ingress Leg (distance variable – fuel dependent)
- Loiter for 30 minutes over target area as Forward Air Controller
- Descend to 300 ft MSL
- Combat for 15 minutes
- Climb to 500 ft MSL
- Egress out 5 nm
- Climb to 3,500 ft MSL
- Return to LHD (Amphibious Assault Ship)
- Descend to 500 ft MSL (pattern altitude)
- Loiter at pattern altitude for 20 minutes (approaches) and recover

TestCobra1					
Total Internal Fuel Available: 1,680 lbs					
Config	Payload (lbs)	Combat Radius (nm)	Burn Rate (lbs/hr)	Fuel Used (lbs)	Fuel Avail. (lbs)
cnfg0	661	70.7	767	1680.1	1680.0
cnfg1	1345	68	783	1680.2	1680.0
cnfg2	425	74.1	755	1680.3	1680.0
cnfg3	889	72.2	766	1680.2	1680.0
cnfg4	515	71.5	763	1680.3	1680.0
cnfg5	1159	68.3	780	1680.3	1680.0
cnfg6	809	70.9	769	1680.3	1680.0
cnfg7	1272	67.6	783	1680.1	1680.0
cnfg8	2640	67.3	791	1680.3	1680.0
cnfg9	784	72.1	765	1680.1	1680.0
cnfg10	800	74.4	759	1680.2	1680.0
cnfg11	1728	70	782	1680.2	1680.0
cnfg12	864	71.6	767	1680.2	1680.0
cnfg13	864	71	769	1680.1	1680.0
cnfg14	636	70.1	768	1680.3	1680.0
cnfg15	1320	67.5	784	1680.2	1680.0

Table 3. **TestCobra1** Output

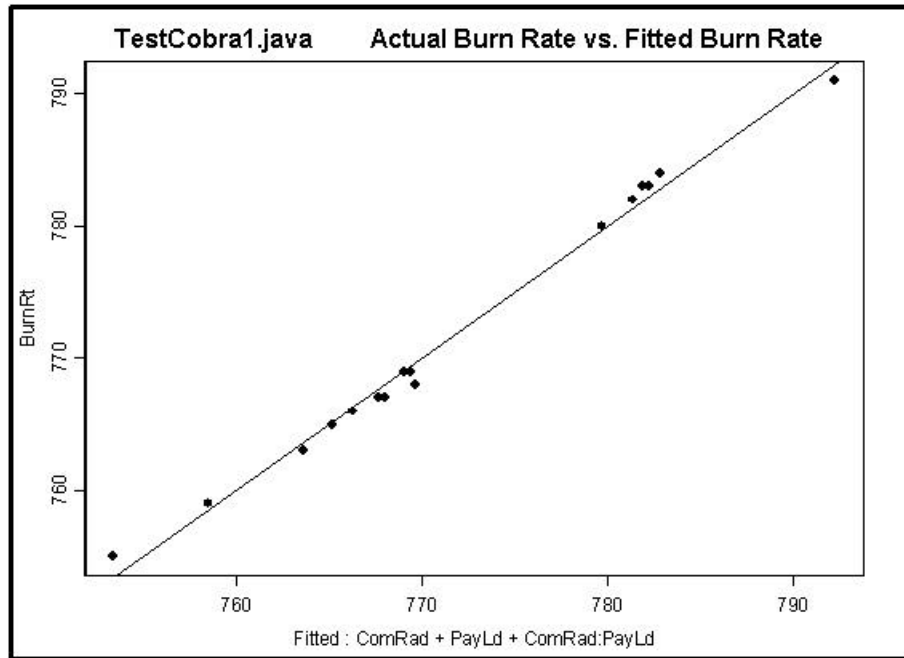


Figure 19. Actual Versus Fitted Values

Regression Statistics						
Multiple R	0.99600785					
R Square	0.992031638					
Adjusted R Square	0.990039547					
Standard Error	1.036833347					
Observations	16					
ANOVA						
	df	SS	MS	F	Significance F	
Regression	3	1606.037219	535.3457398	497.985202	7.48138E-13	
Residual	12	12.90028067	1.075023389			
Total	15	1618.9375				
	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	1048.566283	20.61947747	50.85319375	2.19472E-15	1003.640302	1093.492264
PayLd	-0.084059099	0.021311102	-3.944380747	0.001947441	-0.130491999	-0.037626198
ComRad	-4.074563115	0.294261794	-13.84672831	9.66728E-09	-4.715704476	-3.433421753
PayLd: ComRad	0.001349546	0.000312227	4.322322069	0.000992043	0.000669261	0.00202983

Table 4. Regression Output for **TestCobra1**

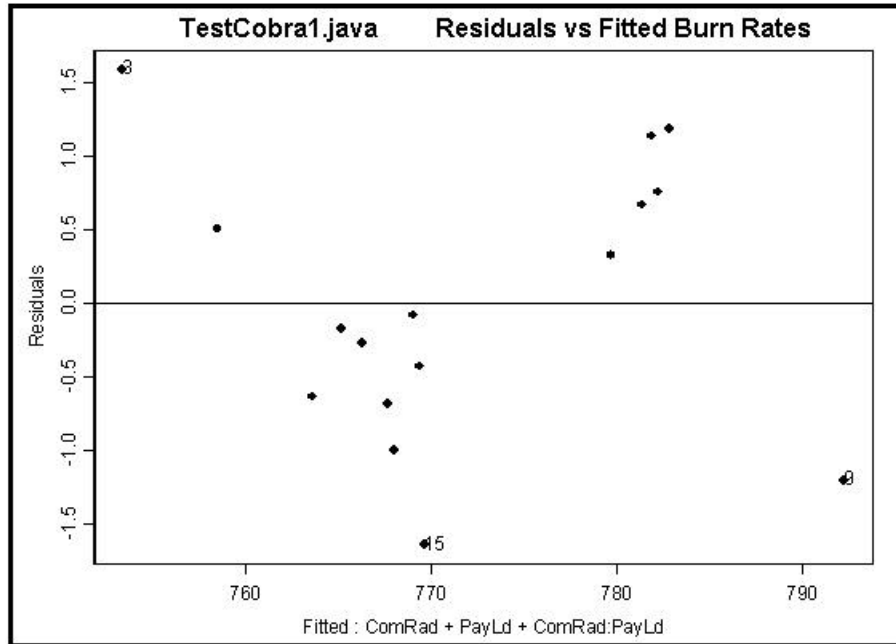


Figure 20. Residual Plot for **TestCobra1.java**.

Regression analysis of the output from **TestCobra1** yields strong correlation between fuel consumption rate (burn rate) and the factors combat radius and payload. The adjusted R^2 value of 0.99 indicates that almost all of the variation in the model is explained by these two factors. The low p-values indicate that payload, combat radius and their interaction (combat radius: payload) all have a statistically significant effect on fuel consumption rate. Therefore, combat radius and payload are good predictors for fuel consumption rate for the AH-1W Cobra.

Model diagnostics for this relationship indicate that this is an appropriate model. In particular, plots of residuals and fitted values for burn rate do not indicate a lack of fit – or heteroscedasticity. Although there is some small nonlinearity from the residual plot, the high coefficient of determination (R^2) along with the very small magnitude of the residuals indicates that the linear fit is more than adequate for the purposes of this thesis. This is further supported by the partial residual plots and Normal probability plots.

B. AH-64A APACHE MISSION PROFILE

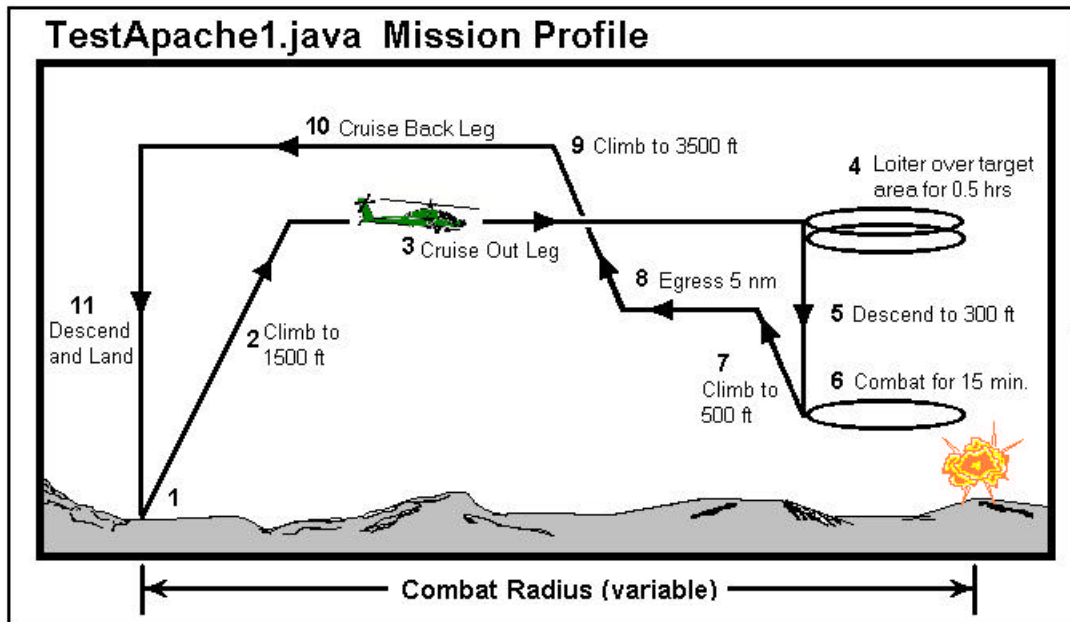


Figure 21. AH-64A Mission Profile

Figure 21 displays a notional mission profile for the AH-64A Apache helicopter. Like **TestCobra1**, this mission profile was written into a Java execution class – **TestApache1**. **TestApache1** examines six different payload configurations for the AH-64A and determines the maximum possible combat radius (in nautical miles) for each. Once again, fuel used is held constant across all configurations (within 0.5 lbs) and combat radius is allowed to vary with fuel consumption rates.

TestApache1 consists of the following flight evolutions:

- Warm up taxi and takeoff
- Climb to 1,500 ft MSL (Mean Sea Level)
- Cruise Out Leg (distance variable – fuel dependent)
- Loiter for 30 minutes over target area as Forward Air Controller
- Descend to 300 ft MSL
- Combat for 15 minutes.
- Climb to 500 ft MSL
- Egress out 5 nm
- Climb to 3,500 ft MSL

- Cruise Back Leg (distance variable – fuel dependent)
- Descend and land

The regression results are displayed in Table 5 and as in the previous case, demonstrate strong correlation between Burn Rate and the factors Combat Radius, Payload and their interaction term. The high adjusted R^2 value indicates that almost all of the variance is explained by this regression model. The low F significance and p-values for each of the coefficients indicate that Combat Radius, Payload and their interaction have a significant effect on fuel consumption rate for the AH-64A Apache helicopter. Figure 22 displays the predicted fuel consumption rates.

<i>Regression Statistics</i>						
Multiple R	0.999900114					
R Square	0.999800237					
Adjusted R Square	0.999500593					
Standard Error	1.076208701					
Observations	6					
<i>ANOVA</i>						
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>	
Regression	3	11593.68355	3864.561183	3336.623384	0.000299629	
Residual	2	2.316450338	1.158225169			
Total	5	11596				
	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	1407.239133	10.85653485	129.6213896	5.95125E-05	1360.527201	1453.951065
PayLd	0.030443437	0.006004277	5.070292199	0.036766735	0.004609102	0.056277772
ComRad	-6.543040582	0.14613748	-44.77318599	0.00049847	-7.171819848	-5.914261316
PayLd: ComRad	-0.00048025	8.76482E-05	-5.47929611	0.031731299	-0.00085737	-0.00010313

Table 5. Regression Output for **TestApache1.java**

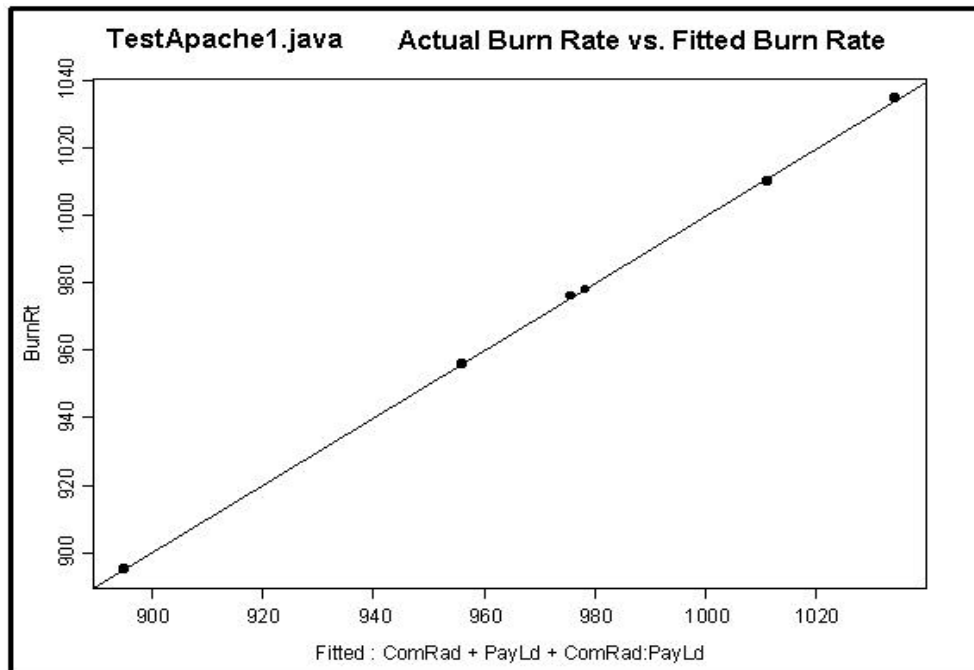


Figure 22. Actual versus Fitted values for **TestApache1**

C. CH-53E SUPER STALLION MISSION PROFILE

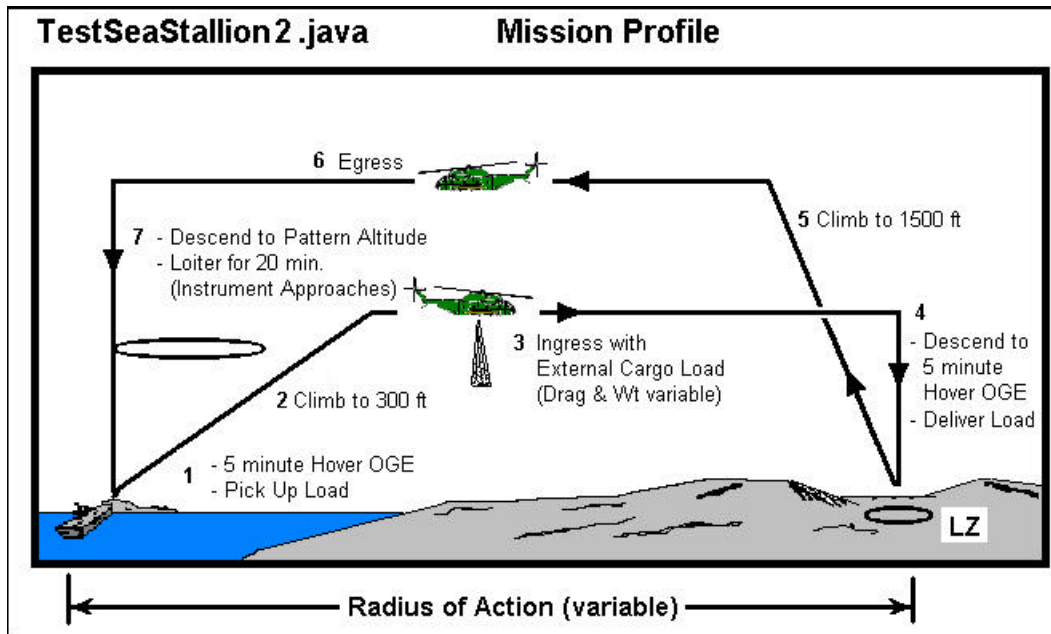


Figure 23. CH53E Mission Profile

Figure 23 displays a notional mission profile for the CH-53E Super Stallion helicopter. This mission profile was written into a Java execution class –

TestSeaStallion2. The execution class examines the CH-53E carrying 30 different external cargo payloads. The weights of these payloads are arbitrarily determined by the author and are well within the limits for external loading per CH-53E NATOPS limitations (maximum external load equals 36,000 lbs [NAVAIR 2002]). The drag areas are determined using a conservative range within the limits specified by the CH-53E NATOPS manual (see Figure 13). Specifically, for this execution, all of the external payload weights are randomly chosen values between 5,000 and 15,000 lbs and the drag areas are randomly chosen values between 33.3 and 100 square feet (see Figure 24).

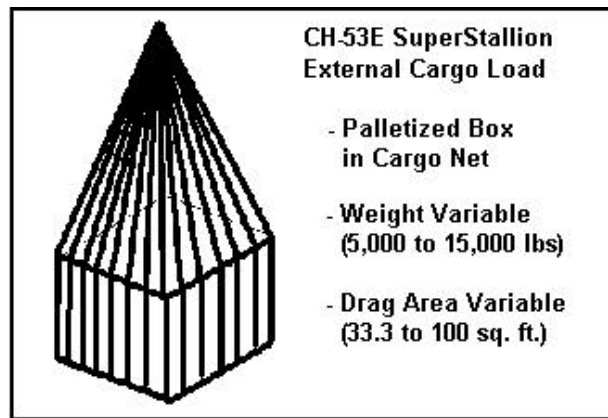


Figure 24. CH-53E External Cargo

For each of these external cargo payloads, the execution class, **TestSeaStallion2** determines the maximum possible radius of action (in nautical miles) for each. Internal fuel carried by each helicopter is constant and equal to the minimum of the maximum possible for all of the CH-53E configurations. Maximum radius of action varies (using the Golden Section Algorithm) according to fuel consumption rate.

TestSeaStallion2 consists of the following flight evolutions:

- Lift into 5 minute hover out of ground effect, pick up external load
- Climb to 300 ft MSL
- Ingress Leg (distance variable – fuel dependent)
- Descend into 5 minute hover OGE, deliver external load
- Climb to 1,500 ft MSL
- Egress Leg
- Descend to 500 ft MSL and loiter over ship for 20 minutes and land

Statistics for the output of **TestSeaStallion2** are displayed in Table 6. The high adjusted R^2 value indicates that the variability in fuel consumption rate is explained well by the factors combat radius, payload, and their interaction. Unlike the previous two cases however, the variability in fuel consumption rates appears to increase with expected fuel consumption rate (see Figure 25). The coefficients from the least squares fit for this model are provided in Figure 6, but the inference results and standard errors – which rely on the assumption of homoscedasticity, are not. Since the goal is to determine the existence of a relationship and not to infer or predict from estimates of that relationship, the regression analysis is not carried further (e.g. variance stabilizing transformations etc. are not explored here).

Regression Statistics						
Multiple R	0.999218347					
R Square	0.998437305					
Adjusted R Square	0.998256994					
Standard Error	2.599009524					
Observations	30					
ANOVA						
	df	SS	MS	F	Significance F	
Regression	3	112211.0406	37403.68018	5537.306882	1.38578E-36	
Residual	26	175.6261131	6.754850504			
Total	29	112386.6667				
	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	4357.566912	73.90225901	58.96392033	3.22942E-29	4205.658544	4509.475281
PayLd	0.038031741	0.007235017	5.256621007	1.70929E-05	0.023159941	0.05290354
ComRad	-2.793196135	0.208854632	-13.37387688	3.65777E-13	-3.222503261	-2.36388901
PayLd: ComRad	-7.79856E-05	2.0614E-05	-3.78313667	0.000820929	-0.000120358	-3.56129E-05

Table 6. Regression Output for **TestSeaStallion2**

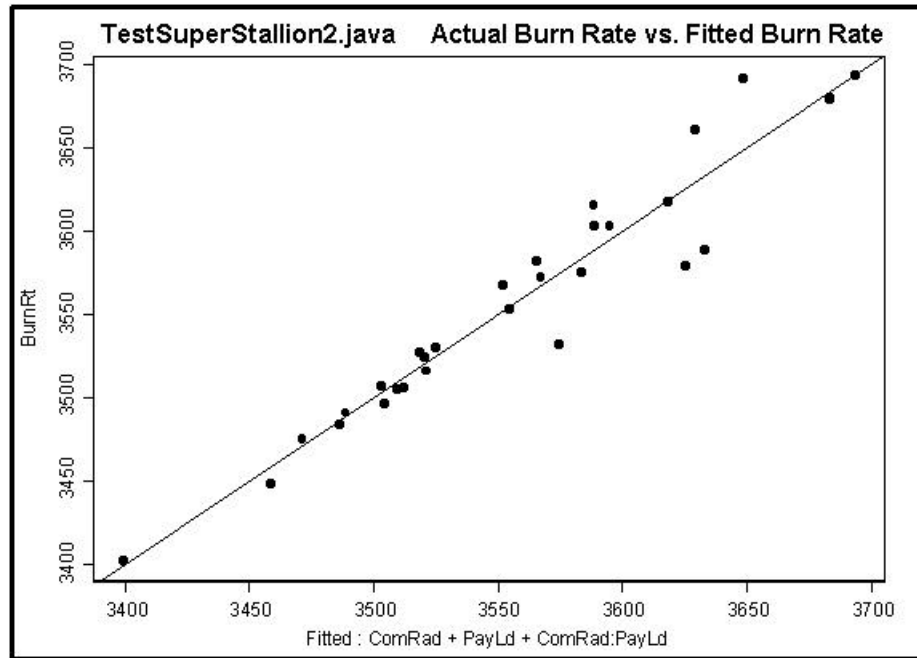


Figure 25. Actual Versus Fitted Values for **TestSeaStallion2**

D. F-14A TOMCAT MISSION PROFILE

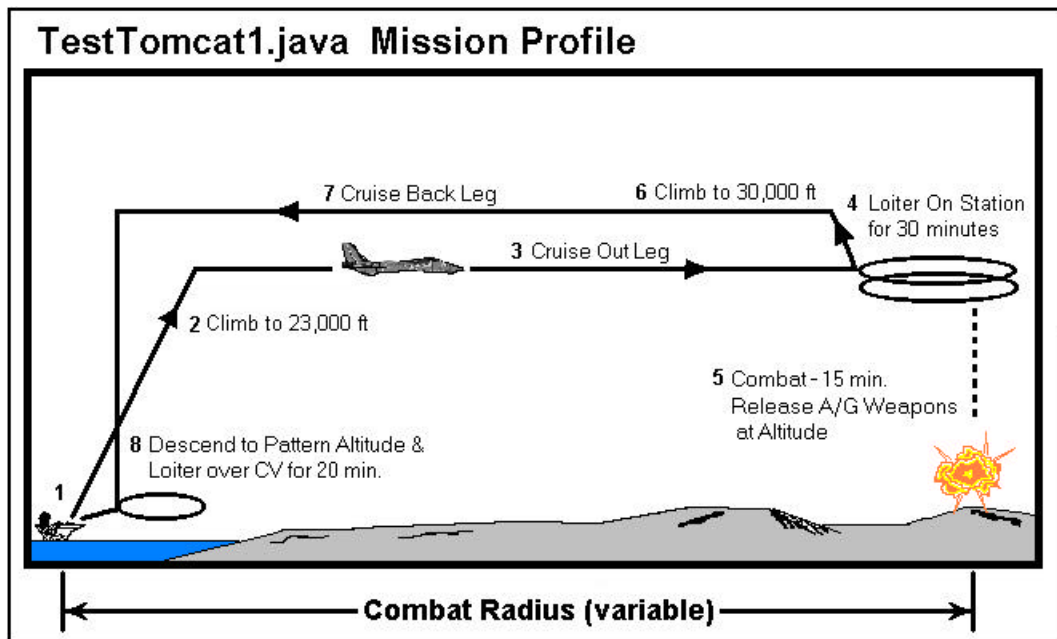


Figure 26. F-14A Mission Profile

Figure 26 displays a notional attack mission profile for the F-14A Tomcat. The execution class, **TestTomcat1**, utilizes this mission profile to examine thirty different

weapons configurations of the F-14A. These configurations are displayed in Table 7 (refer to Figure 16 for fuselage locations of weapons stations). Maximum combat radius is determined based on fuel consumption rates for each of the thirty different F-14A weapons configurations. Drag index information and payload weights were determined from the F-14 Naval Weapons Publication, NWP 3-22.5-F14A/B/D, Volume III. The regression statistics and fitted values versus actual values plot are provided in Table 8 and Figure 27 respectively.

Configuration	F-14A Weapon Station							
	1	2	3	4	5	6	7	8
config00	AIM-7/9	Empty	GBU-16	Empty	Empty	GBU-16	Empty	AIM-9LAN
config01	AIM-7/9	Empty	GBU-16	GBU-16	Empty	GBU-16	Empty	AIM-9LAN
config02	AIM-7/9	Empty	GBU-16	GBU-16	GBU-16	GBU-16	Empty	AIM-9LAN
config03	AIM-7/9	Empty	GBU-16	GBU-16	CBU-100	GBU-16	Empty	AIM-9LAN
config04	AIM-7/9	Empty	GBU-16	CBU-100	CBU-100	GBU-16	Empty	AIM-9LAN
config05	AIM-7/9	Empty	GBU-16	CBU-99	CBU-99	GBU-16	Empty	AIM-9LAN
config06	AIM-54	Empty	GBU-16	CBU-99	GBU-16	GBU-16	Empty	AIM-9LAN
config07	AIM-7/9	Empty	GBU-24	CBU-100	CBU-100	GBU-16	Empty	AIM-9LAN
config08	AIM-7/9	Empty	GBU-24	Empty	GBU-24	Empty	Empty	AIM-9LAN
config09	AIM-7/9	Empty	GBU-24	GBU-16	GBU-24	Empty	Empty	AIM-9LAN
config10	AIM-7/9	Empty	GBU-24	GBU-16	GBU-24	GBU-16	Empty	AIM-9LAN
config11	AIM-7/9	Empty	GBU-12	Empty	ADM-141	GBU-12	Empty	AIM-9LAN
config12	AIM-7/9	Empty	GBU-12	BLU-111	Empty	GBU-12	Empty	AIM-9LAN
config13	AIM-7/9	Empty	GBU-12	BLU-111	ADM-141	GBU-12	Empty	AIM-9LAN
config14	AIM-7/9	Empty	GBU-12	Empty	BLU-110	GBU-12	Empty	AIM-9LAN
config15	AIM-54	Empty	GBU-12	MK-83	ADM-141	GBU-12	Empty	AIM-9LAN
config16	AIM-7/9	Empty	GBU-12	CBU-99	CBU-100	GBU-12	Empty	AIM-9LAN
config17	AIM-7/9	Empty	GBU-12	BLU-111	BLU-111	GBU-12	Empty	AIM-9LAN
config18	AIM-54	Empty	GBU-16	GBU-12	GBU-12	GBU-16	Empty	AIM-9LAN
config19	AIM-7/9	Empty	GBU-12	ADM-141	BLU-110	BLU-111	Empty	AIM-9LAN
config20	AIM-54	Empty	GBU-12	BLU-111	BLU-111	GBU-12	Empty	AIM-9LAN
config21	AIM-7/9	Empty	GBU-16	GBU-12	GBU-12	Empty	Empty	AIM-9LAN
config22	AIM-7/9	Empty	GBU-12	ADM-141	BLU-110	GBU-12	Empty	AIM-9LAN
config23	AIM-7/9	Empty	GBU-12	BLU-110	BLU-110	GBU-12	Empty	AIM-9LAN
config24	AIM-7/9	Empty	GBU-12	BLU-110	GBU-16	GBU-12	Empty	AIM-9LAN
config25	AIM-7/9	Empty	GBU-10	CBU-99	GBU-12	GBU-10	Empty	AIM-9LAN
config26	AIM-7/9	Empty	GBU-10	GBU-12	GBU-16	GBU-10	Empty	AIM-9LAN
config27	AIM-7/9	Empty	GBU-12	GBU-12	GBU-16	GBU-12	Empty	AIM-9LAN
config28	AIM-7/9	Empty	GBU-10	GBU-12	GBU-12	GBU-10	Empty	AIM-9LAN
config29	AIM-7/9	Empty	GBU-12	GBU-12	GBU-12	GBU-12	Empty	AIM-9LAN

NOTES: AIM 7/9 at Sta. 1 indicates AIM-7 and AIM-9 missiles. AIM-9LAN at Sta 8 indicates AIM - 9 missile and LANTIRN pod. Stations 2 and 7 are used for external fuel tanks only (when carried).

Table 7. F-14A Weapons Configurations Used in **TestTomcat1**

The regression output for **TestTomcat1** again indicates a strong relationship between the factors combat radius, payload, and their interaction with fuel consumption rate. A strong adjusted R^2 value, very low F significance as well as low p-values for each of the coefficients in the regression model are all indicators of a strong relationship.

Regression Statistics						
Multiple R	0.999952079					
R Square	0.99990416					
Adjusted R Square	0.999893102					
Standard Error	5.708379319					
Observations	30					

ANOVA					
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	3	8839156.275	2946385.425	90419.87647	2.4084E-52
Residual	26	847.2254557	32.58559445		
Total	29	8840003.5			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	10042.06402	50.84213765	197.5145908	7.8655E-43	9937.556437	10146.5716
PayLd	0.117137511	0.004118595	28.44113527	4.07445E-21	0.108671613	0.12560341
ComRad	-13.46124418	0.180094057	-74.7456324	7.0214E-32	-13.83143306	-13.09105531
PayLd: ComRad	-0.000439663	1.76504E-05	-24.90951965	1.13748E-19	-0.000475944	-0.000403382

Table 8. Regression Output for **TestTomcat1**

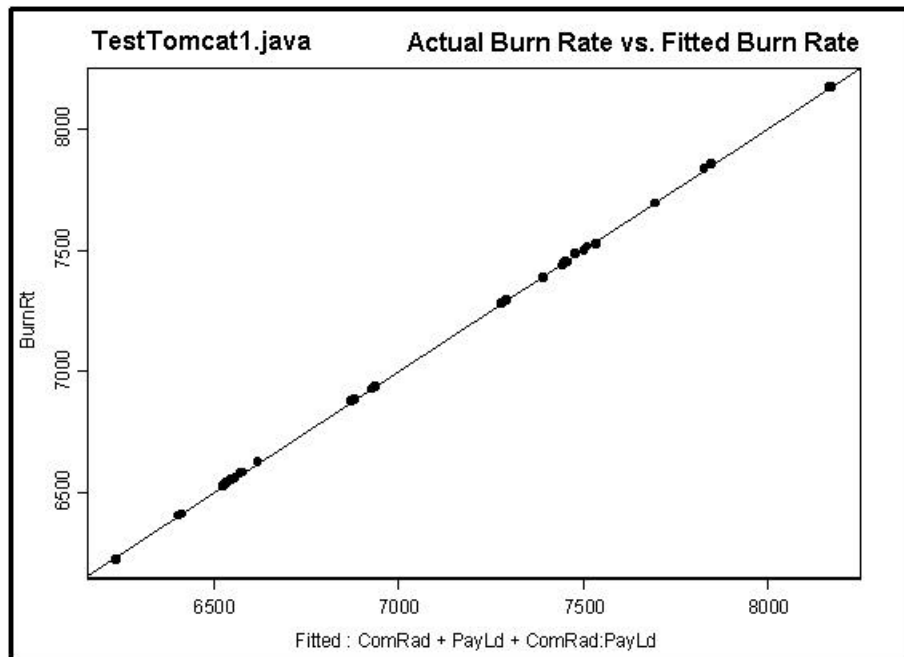


Figure 27. Actual Versus Fitted Values for **TestTomcat1**

E. F/A-18C HORNET MISSION PROFILES

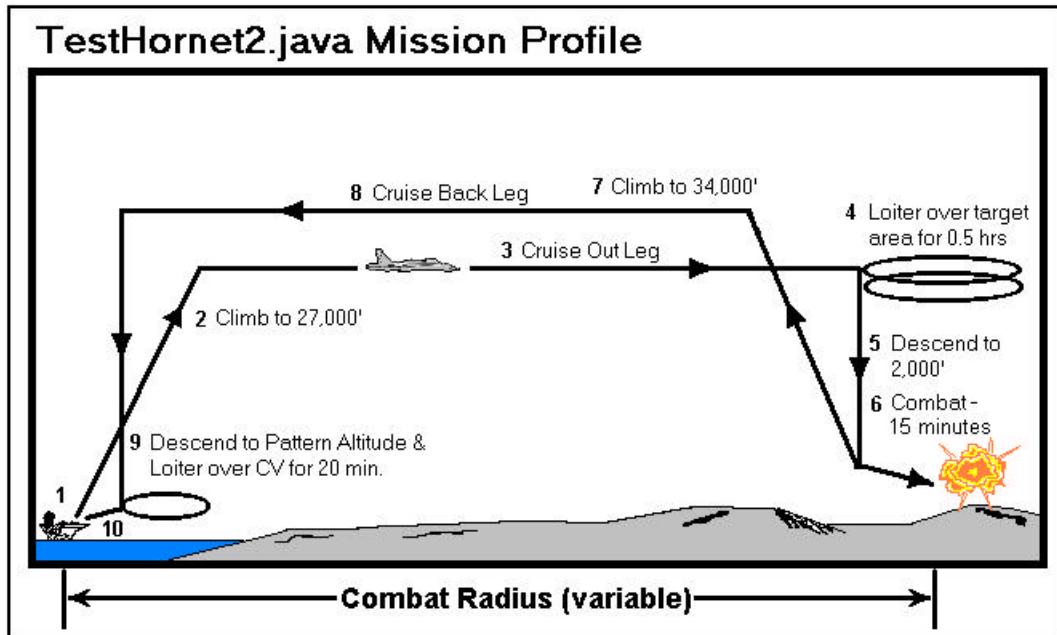


Figure 28. F/A-18C Mission Profile (Combat Radius Variable)

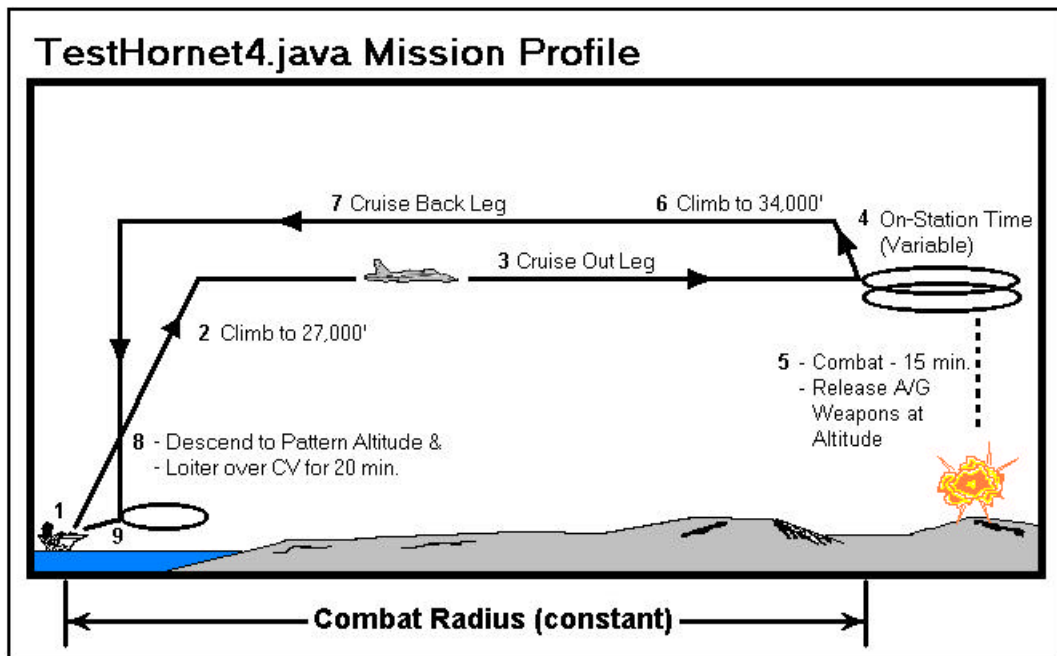


Figure 29. F/A-18C Mission Profile (On-Station Endurance Variable)

Figures 28 and 29 display two different mission profiles for the F/A-18C aircraft. In Figure 28, the maximum combat radius is allowed to vary according to fuel consumption rate and, in Figure 29, the maximum on-station endurance is allowed to

vary according to fuel consumption. Thirty different weapons configurations of the F/A-18C were used in each of these two mission profiles. These weapon configurations are detailed in Table 9. Refer to Figure 17 for fuselage locations of the respective weapon stations on the F/A-18C Hornet.

Configuration	F/A-18C Weapon Station								
	9	8	7	6	5	4	3	2	1
config00	AIM-9	AGM-65	AGM-65	ASQ-173	MK-82	AAS-38	AGM-65	AGM-65	AIM-9
config01	AIM-9	AGM-84E	AGM-84E	ASQ-173	MK-82	AAS-38	AGM-84E	AWW-9B	AIM-9
config02	AIM-9	AGM-84E	AGM-88	ASQ-173	MK-83	AAS-38	AGM-88	AWW-9B	AIM-9
config03	AIM-9	GBU-10	AGM-88	ASQ-173	MK-83	AAS-38	AGM-88	GBU-10	AIM-9
config04	AIM-9	GBU-24	AGM-88	ASQ-173	MK-83	AAS-38	AGM-88	GBU-24	AIM-9
config05	AIM-9	GBU-24	GBU-16	ASQ-173	MK-20	AAS-38	GBU-16	GBU-24	AIM-9
config06	AIM-9	AGM-62A	GBU-16	ASQ-173	MK-20	AAS-38	GBU-16	AWW-9B	AIM-9
config07	AIM-9	AGM-62A	GBU-12	ASQ-173	MK-20	AAS-38	GBU-12	AWW-9B	AIM-9
config08	AIM-9	GBU-12	GBU-12	ASQ-173	MK-20	AAS-38	GBU-12	GBU-12	AIM-9
config09	AIM-9	GBU-16	GBU-16	ASQ-173	MK-20	AAS-38	GBU-16	GBU-16	AIM-9
config10	AIM-9	GBU-10	GBU-16	ASQ-173	MK-20	AAS-38	GBU-16	GBU-10	AIM-9
config11	AIM-9	GBU-10	AGM-88	ASQ-173	MK-20	AAS-38	AGM-88	GBU-10	AIM-9
config12	AIM-9	GBU-24	AGM-88	ASQ-173	MK-20	AAS-38	AGM-88	GBU-24	AIM-9
config13	AIM-9	WALLEYE-II	AGM-88	ASQ-173	MK-20	AAS-38	AGM-88	AWW-13	AIM-9
config14	AIM-9	WALLEYE-II	GBU-12	ASQ-173	MK-20	AAS-38	GBU-12	AWW-13	AIM-9
config15	AIM-9	AGM-84E	GBU-12	ASQ-173	MK-20	AAS-38	GBU-12	AWW-13	AIM-9
config16	AIM-9	AGM-84E	AGM-88	ASQ-173	Empty	AAS-38	AGM-88	AWW-13	AIM-9
config17	AIM-9	AGM-84E	GBU-12	ASQ-173	Empty	AAS-38	GBU-12	AWW-13	AIM-9
config18	AIM-9	GBU-10	GBU-12	ASQ-173	Empty	AAS-38	GBU-12	GBU-10	AIM-9
config19	AIM-9	AGM-65	AGM-65	ASQ-173	Empty	AAS-38	AGM-65	AGM-65	AIM-9
config20	AIM-9	AGM-84E	AGM-65	ASQ-173	Empty	AAS-38	AGM-65	AWW-13	AIM-9
config21	AIM-9	GBU-24	AGM-65	ASQ-173	Empty	AAS-38	AGM-65	GBU-24	AIM-9
config22	AIM-9	GBU-24	MK-84	ASQ-173	Empty	AAS-38	MK-84	GBU-24	AIM-9
config23	AIM-9	GBU-24	MK-83	ASQ-173	Empty	AAS-38	MK-83	GBU-24	AIM-9
config24	AIM-9	GBU-24	MK-20	ASQ-173	Empty	AAS-38	MK-20	GBU-24	AIM-9
config25	AIM-9	GBU-10	MK-20	ASQ-173	Empty	AAS-38	MK-20	GBU-10	AIM-9
config26	AIM-9	GBU-12	MK-20	ASQ-173	Empty	AAS-38	MK-20	GBU-12	AIM-9
config27	AIM-9	GBU-12	MK-82	ASQ-173	Empty	AAS-38	MK-82	GBU-12	AIM-9
config28	AIM-9	MK-83	MK-83	ASQ-173	MK-83	AAS-38	MK-83	MK-83	AIM-9
config29	AIM-9	MK-82	MK-82	ASQ-173	MK-82	AAS-38	MK-82	MK-82	AIM-9

Table 9. F/A-18C Weapon Configurations for **TestHornet2** and **TestHornet4**

1. **TestHornet2** Mission Profile

The mission profile for **TestHornet2** proceeds as follows:

- Warm up, taxi and takeoff
- Climb to 27,000 ft MSL
- “Cruise Out” Leg (distance variable – fuel dependent)
- Loiter over Target Area for 30 minutes (on-station time held constant)
- Descend to 2,000 ft MSL
- Weapons Delivery and Combat (15 min. maximum continuous power)
- Climb to 31,000 ft MSL

- “Cruise Back” Leg
- Descend to 1,500 ft MSL pattern altitude, loiter for 20 minutes
- Recover aboard CVN

TestHornet2 varies the combat radius according to fuel consumption rate in the same manner as **TestCobra1**, **TestApache1**, **TestSeaStallion2**, and **TestTomcat1**. The regression output of **TestHornet2** is displayed in Table 10.

Regression Statistics	
Multiple R	0.996402817
R Square	0.992818574
Adjusted R Square	0.991989948
Standard Error	45.49382829
Observations	30

ANOVA					
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	3	7439393.568	2479797.856	1198.150331	5.63544E-28
Residual	26	53811.89873	2069.688413		
Total	29	7493205.467			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	6836.64233	183.6390776	37.2286902	4.37446E-24	6459.166552	7214.118107
PayLd	0.196269871	0.020029563	9.799009112	3.23018E-10	0.155098487	0.237441254
ComRad	-6.539287846	1.290832285	-5.065946925	2.82401E-05	-9.192633348	-3.885942345
PayLd: ComRad	-0.001115808	0.000175699	-6.35069094	1.00257E-06	-0.001476962	-0.000754654

Table 10. Regression Output for **TestHornet2**

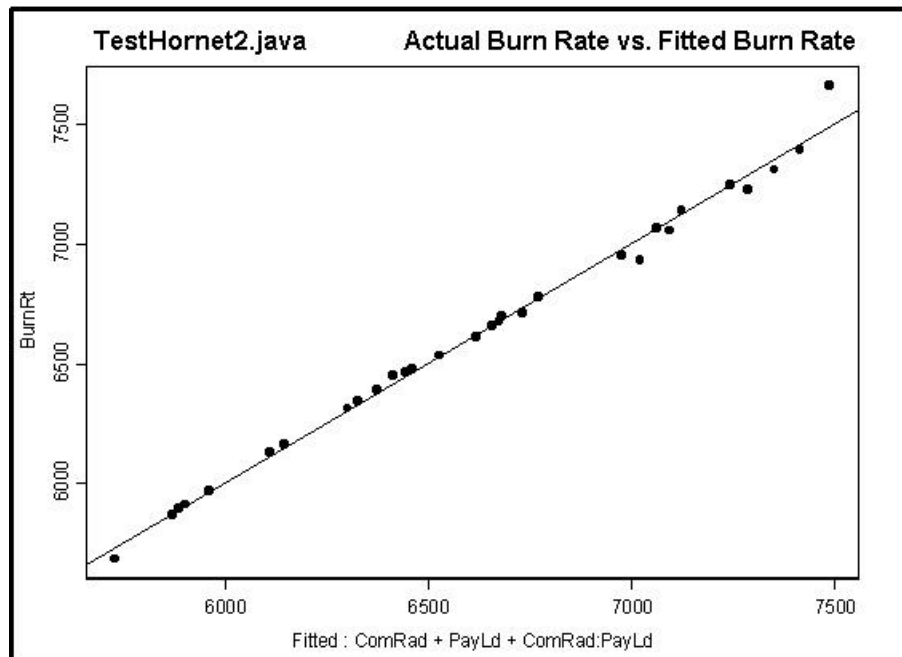


Figure 30. Actual Versus Fitted Values for **TestHornet2**

The strong adjusted R^2 value, low model F-significance and low coefficient p-values indicate a strong relationship of the factors combat radius, payload and their interaction with fuel consumption rate.

2. **TestHornet4 Mission Profile**

TestHornet4 is a mission profile which determines the relationship between on-station endurance and payload with fuel consumption rate. In some aircraft acquisitions, on-station endurance is used as a KPP vice combat radius. Therefore, it is useful to examine the relationship that exists between these factors. **TestHornet4** utilizes a similar mission profile to **TestHornet2** except that the combat radius is held constant and the on-station loiter time (over the target area) is allowed to vary according to fuel consumption rate.

The mission profile for **TestHornet4** proceeds as follows:

- Warm up, taxi and takeoff.
- Climb to 27,000 ft MSL.
- Cruise out 150 nautical miles.
- Loiter over Target Area (on-station endurance time dependent on fuel consumption rate).
- Weapons Delivery and Combat (15 minutes at maximum continuous power, air-to-ground weapons delivered at altitude).
- Climb to 34,000 ft MSL.
- “Cruise Back” Leg.
- Descend to 1,500 ft MSL pattern altitude, loiter for 20 minutes.
- Recover aboard CVN.

The regression output for **TestHornet4** is displayed in Table 11. The high adjusted R^2 value, low model F-significance and low coefficient p-values all indicate that there is a strong relationship between fuel consumption rate and the factors on-station endurance, payload and their interaction. Additionally, the fitted values plot (Figure 31) – even with slight non-linearity of residuals – is further evidence of a strong relationship.

Regression Statistics	
Multiple R	0.999022599
R Square	0.998046153
Adjusted R Square	0.997820709
Standard Error	20.29917499
Observations	30

ANOVA					
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	3	5472555.198	1824185.066	4427.026494	2.52828E-35
Residual	26	10713.46914	412.0565053		
Total	29	5483268.667			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	6394.46281	110.6236832	57.8037417	5.39358E-29	6167.072424	6621.853196
PayLd	0.093770794	0.010391223	9.024037873	1.71976E-09	0.072411314	0.115130273
OnStaT	-1217.938311	74.00034231	-16.45854969	2.88225E-15	-1370.048293	-1065.828329
PayLd: OnStaT	-0.067244505	0.007721899	-8.708286124	3.47657E-09	-0.083117106	-0.051371904

Table 11. Regression Output for **TestHornet4**

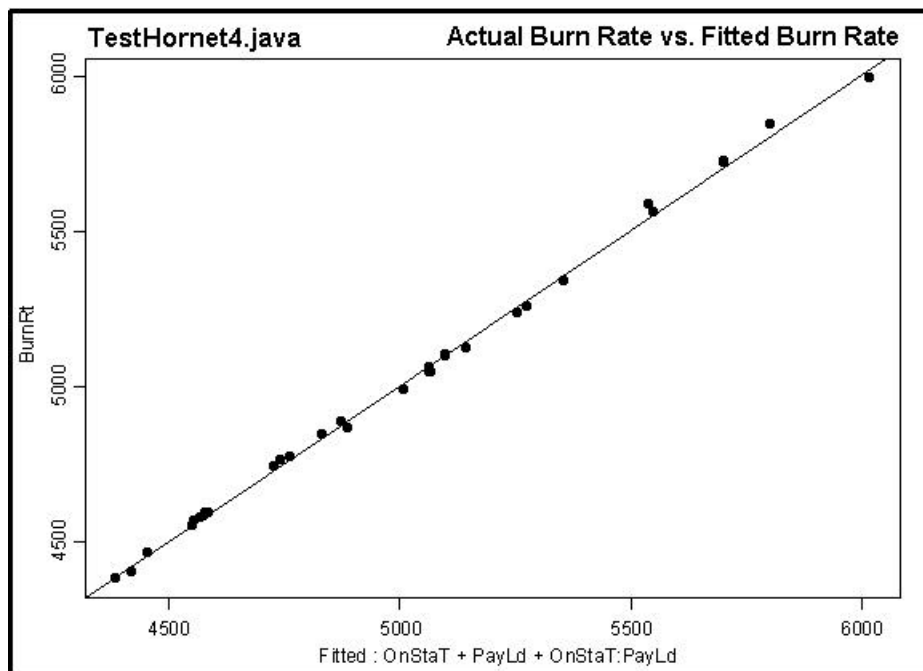


Figure 31. Actual Versus Fitted Values for **TestHornet4**

F. SEAHAWK HELICOPTER MISSION PROFILES

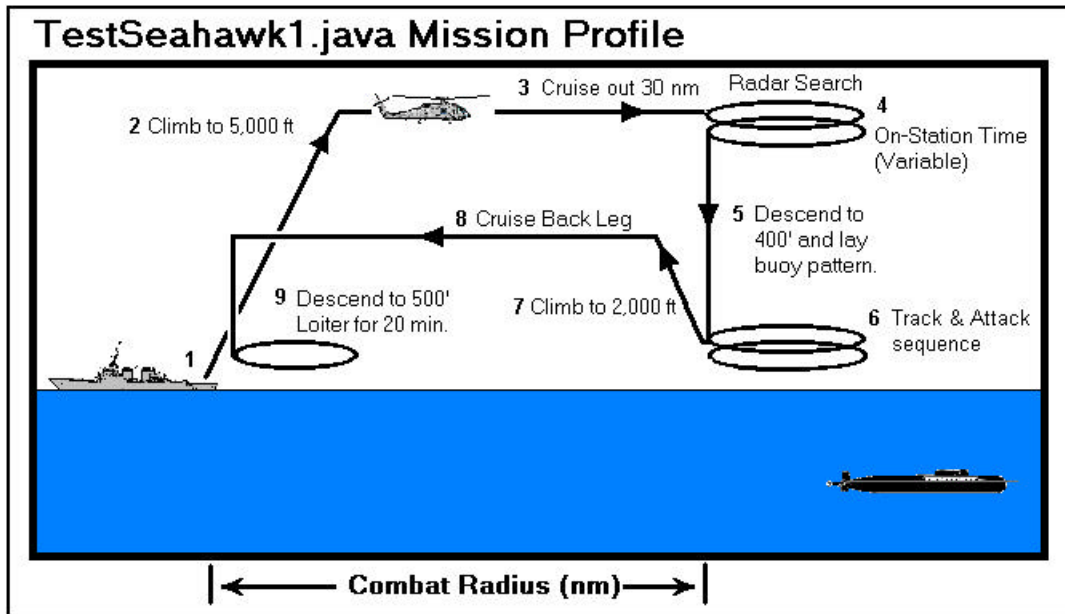


Figure 32. SH60B Seahawk Mission Profile (On-Station Endurance Variable)

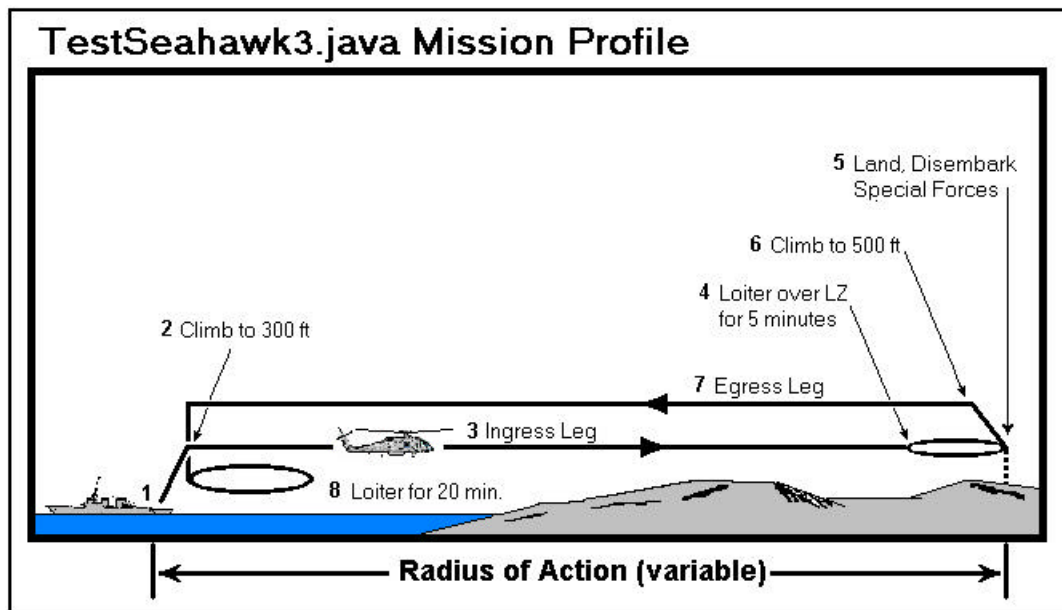


Figure 33. HH60H Seahawk Mission Profile (Radius of Action Variable)

Figures 32 and 33 present two different mission profiles for two variants for the U.S. Navy's H-60 Seahawk helicopter - the SH-60B and the HH-60H.

1. **TestSeahawk1 and TestSeahawk2 Mission Profiles**

Figure 32 displays a typical mission profile for the SH-60B Seahawk helicopter. The SH-60B is the U.S. Navy's primary rotary-wing asset operating from cruisers and destroyers utilized for Undersea Warfare (USW) and Anti-Ship Surveillance and Targeting (ASST). **TestSeahawk1** is a Java execution class that examines sixteen different weapon configurations of the SH-60B flying the mission profile from Figure 32. These weapon fuselage locations and configurations are displayed in Figure 34. The fuel used for each helicopter configuration in **TestSeahawk1** is held constant and equals the minimum of the maximum possible capacity with the heaviest weapon payload (in this case config00). Therefore, each configuration uses the same amount of fuel. The combat radius varies based on fuel consumption rate and the amount of fuel available.

The **TestSeahawk1** and **TestSeahawk2** mission profile proceeds as follows:

- 10 minute warm up on CG flight deck and takeoff
- Climb to 5,000 ft MSL
- Cruise out 30 nautical miles conducting radar search
- Loiter over Search Area (on-station endurance time dependent on fuel consumption rate)
- Descend to 400 ft and deploy sonobuoy pattern for 18 minutes
- Track and attack sequence on submarine contact at maximum continuous power for 15 minutes
- Climb to 2,000 ft MSL
- Return leg 30 nm
- Descend to 500 ft MSL pattern altitude, loiter for 20 minutes
- Recover aboard CG

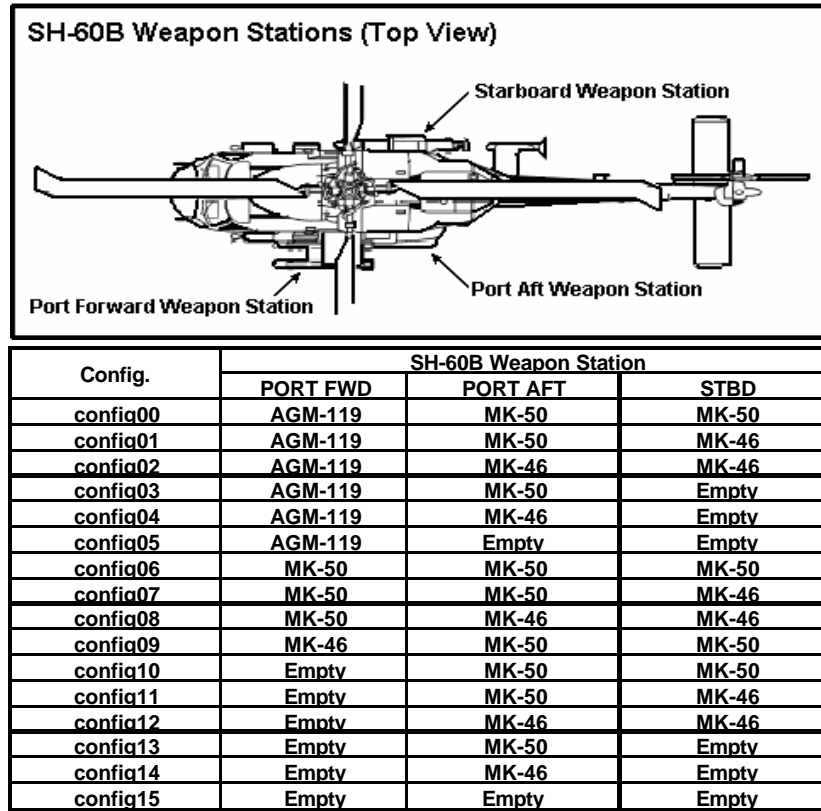


Figure 34. SH-60B Weapon Stations and Configurations for **TestSeahawk1** and **TestSeahawk2**

Table 12 is a graph of the regression output from **TestSeahawk1** showing fuel consumption rate versus the factors on-station time, payload and their interaction. The adjusted R^2 is extremely strong (nearly 1.0). The results of **TestSeahawk1** are remarkably linear. Figures 35 and 36 show fuel consumption rate plotted against the factors on-station endurance and payload respectively. In each case, the linear relationship is quite strong. It is possible that these relationships are perfectly linear and that the residual errors are likely attributable to the model construction - when the NATOPS performance charts were transcribed into matrices for the Java class **SH60B**.

<i>Regression Statistics</i>						
Multiple R	0.999967377					
R Square	0.999934754					
Adjusted R Square	0.999918443					
Standard Error	0.300811389					
Observations	16					
ANOVA						
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>	
Regression	3	16641.35165	5547.117217	61302.58578	2.26235E-25	
Residual	12	1.085849899	0.090487492			
Total	15	16642.4375				
	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	1433.527874	66.37766599	21.59653933	5.67958E-11	1288.903367	1578.152382
PayLd	0.019740436	0.008691548	2.271222258	0.042344433	0.000803181	0.038677691
OnStaT	-376.3275653	87.91954081	-4.280363181	0.001068325	-567.8877856	-184.7673451
PayLd: OnStaT	-0.016731911	0.004015173	-4.167170624	0.001306004	-0.025480221	-0.007983601

Table 12. Regression for **TestSeahawk1**

Recall that the SH-60B performance charts do not account for drag. The small total change in the cross-sectional area as a result of the various weapon configurations for the SH-60B combined with the subsonic drag characteristics associated with the helicopter flight envelope apparently have a very minimal impact on the overall drag characteristics of the SH-60B. This may justify the lack of drag information in the SH-60B NATOPS performance charts. Consequently, changes in payload and on-station endurance are likely directly proportional to fuel consumption rates.

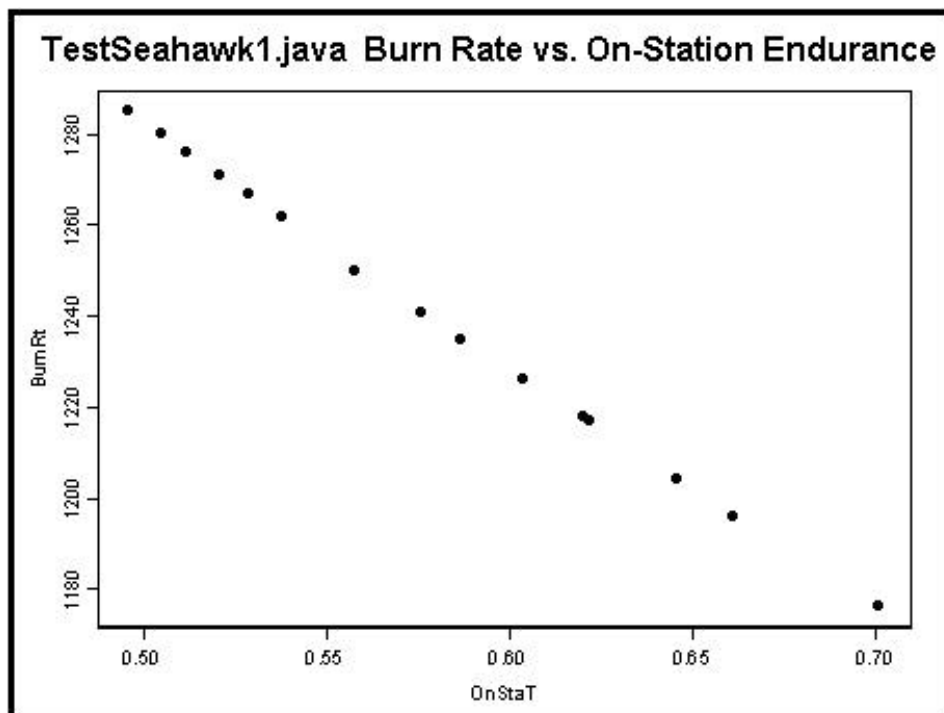


Figure 35. SH-60B Fuel Consumption Rate Versus On-Station Endurance

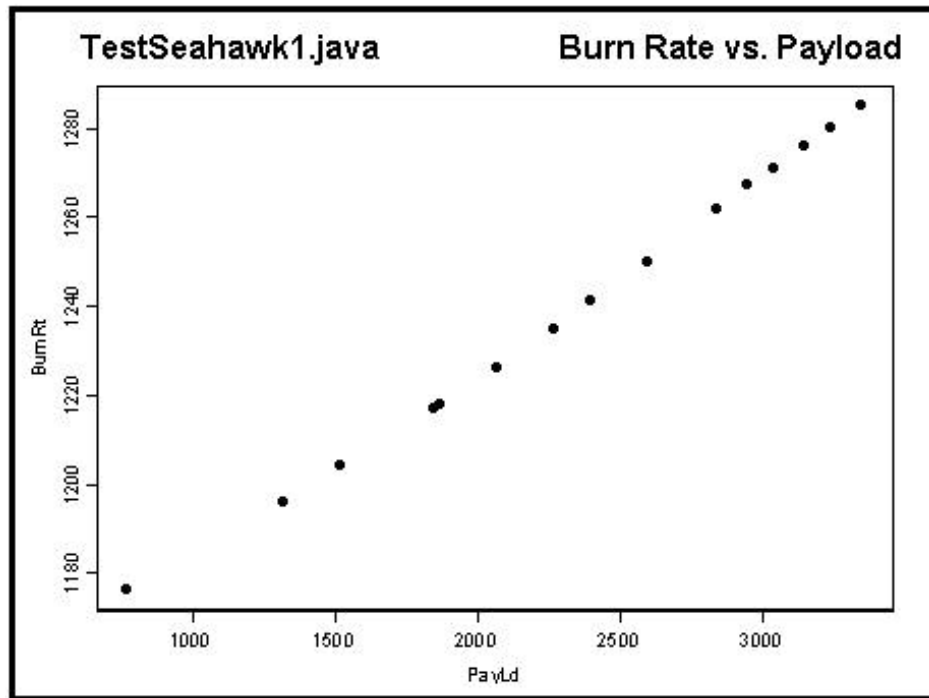


Figure 36. SH-60B Fuel Consumption Rate Versus Payload

TestSeahawk2 uses the same mission profile and weapon configurations as used in **TestSeahawk1**. However, **TestSeahawk2** allows each configuration to carry the maximum amount of internal fuel possible - up to the NATOPS limit (fuel is limited by maximum internal capacity or by maximum gross weight at takeoff). The results are interestingly different from those of **TestSeahawk1** and are displayed in Figure 37.

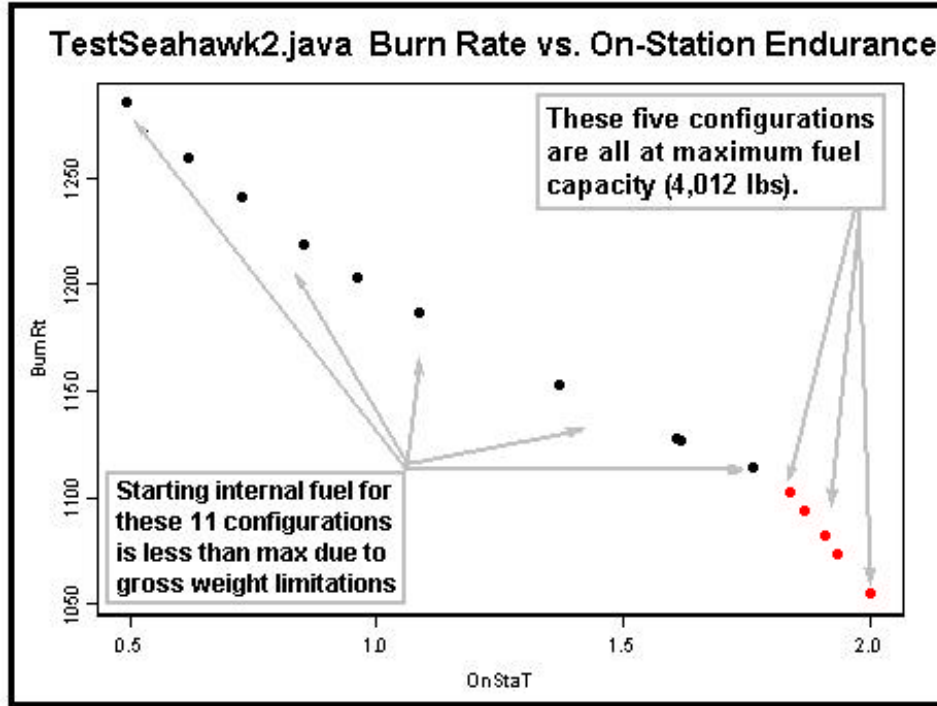


Figure 37. Fuel Consumption Versus On-Station Endurance

The relationship between fuel consumption rate and on-station endurance appears to follow a polynomial fit when maximum internal fuel at takeoff varies according to payload - until the maximum fuel capacity is reached. The last five configurations shown in Figure 36 all have the same amount of fuel at takeoff. Their payloads are light enough to allow them to launch with a full tank of fuel. Notice that their relationship to fuel consumption rate is very nearly linear.

The introduction of a binary indicator variable to the model for fuel consumption rate vs. on-station endurance for **TestSeahawk2** allows for a change in the linear fit for the last five configurations in Figure 37. A binary variable, *MaxFuel*, is added to the regression model such that *MaxFuel* equals one if the configuration launches with maximum internal fuel (4,012 lbs) and zero otherwise. The regression model is:

$$BurnRt = \beta_0 + \beta_1 * OnStaT + \beta_2 * MaxFuel + \beta_3 * OnStaT * MaxFuel$$

where: *OnStaT* = on-station endurance (hrs)

MaxFuel = binary variable indicating maximum internal fuel

β_0 = regression model intercept

β_1, β_2 and β_3 = regression model coefficients

MaxFuel is binary, therefore, β_2 and β_3 act as intercept and slope modifiers when fuel is at maximum capacity. From Figure 38, note that the piecewise linear fit does not completely explain the relationship in Figure 37. However the very high adjusted R^2 of 0.99 indicates that it does explain most of the variability in fuel consumption. Thus, taking into account the binary variable, MaxFuel, there is a very strong relationship between fuel consumption rate and on-station endurance.

Regression Statistics						
Multiple R	0.997203586					
R Square	0.994414993					
Adjusted R Square	0.993018741					
Standard Error	6.16659699					
Observations	16					

ANOVA						
	df	SS	MS	F	Significance F	
Regression	3	81248.67698	27082.89233	712.2031824	8.87883E-14	
Residual	12	456.3230212	38.02691844			
Total	15	81705				

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	1336.270105	4.94588943	270.1779173	4.4484E-24	1325.493938	1347.046272
OnStaT	-131.0473478	4.121495166	-31.79606975	5.90698E-13	-140.0273142	-122.0673814
MaxFuel	300.6253022	91.74717325	3.276671003	0.006620846	100.7253874	500.5252169
OnStaT:MaxFuel	-159.8279569	48.07543859	-3.324524156	0.006059214	-264.5753375	-55.08057629

Table 13. **TestSeahawk2** Regression Output with Indicator Variable

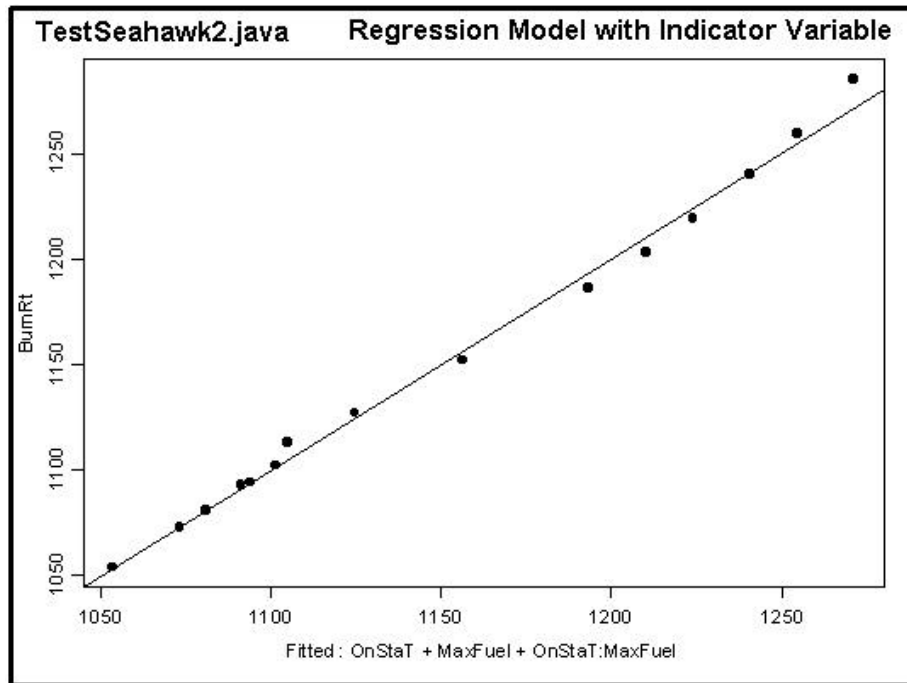


Figure 38. **TestSeahawk2** Actual Versus Fitted Values

2. **TestSeahawk3 and TestSeahawk4 Mission Profiles**

Figure 33 displays a typical mission profile for the HH-60H. The HH-60H helicopter is the Navy's primary rotary-wing asset utilized for Naval Special Warfare (NSW) and Combat Search and Rescue (CSAR). The Java execution classes **TestSeahawk3**, and **TestSeahawk4** examine the HH-60H carrying sixteen different internal payloads.

The **TestSeahawk3** and **TestSeahawk4** mission profile proceeds as follows:

- 10 minute warm up on CG flight deck and takeoff
- Climb to 300 ft MSL
- Ingress Leg (radius of action dependent on fuel consumption rate)
- Loiter over Landing Zone (LZ) for 5 minutes
- Descend, briefly land for Special Forces insertion.
- Climb to 500 ft MSL
- Return leg
- Loiter for 20 minutes
- Recover aboard CG

The payload weights start at 200 lbs and continue incrementally to a maximum of 3,200 lbs. For each payload, the maximum radius of action is determined based on fuel consumption rate. The execution class **TestSeahawk3** holds fuel constant across all sixteen HH-60H configurations and is equal to the minimum of the maximum fuel allowed for each configuration. In this case the sixteenth configuration carrying the heaviest internal payload (3,200 lbs) has the limiting fuel capacity. Figures 38 and 39 display fuel consumption rate versus each of the factors payload and radius of action. Once again the results are highly linear. In this case, since the payload carried by the HH-60H helicopter is an internal load, the cross-sectional drag area of the aircraft is constant across all configurations. Also - like the SH-60B - the HH-60H NATOPS performance charts do not account for changes in drag. These plots (Figures 38 and 39) demonstrate a strong relationship between fuel consumption rates and the factors payload and radius of action. Therefore, any changes in payload and radius of action are likely proportional to changes in fuel consumption rates for the HH-60H.

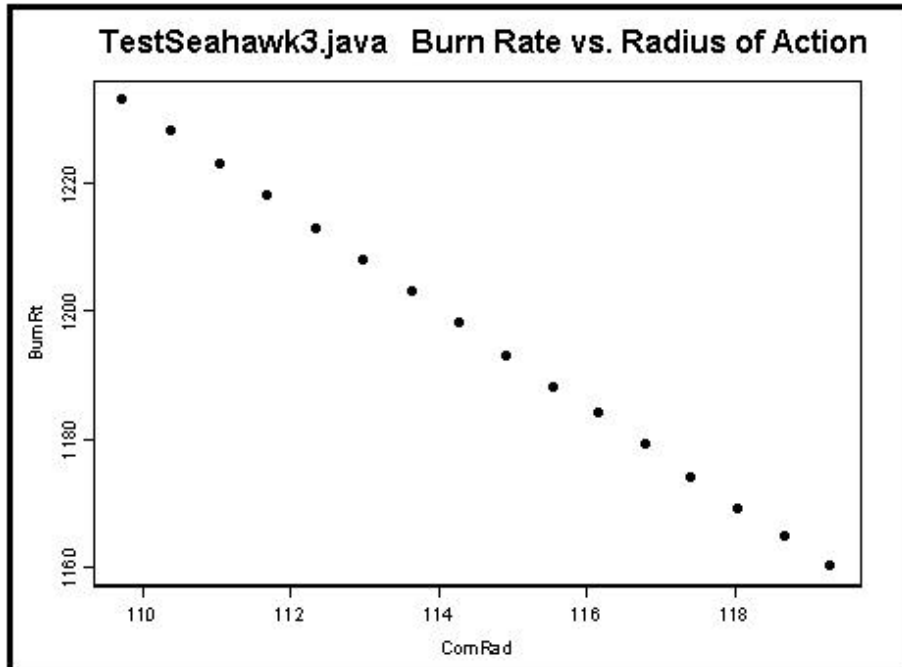


Figure 39. **TestSeahawk3** Fuel Consumption Rate Versus Radius of Action

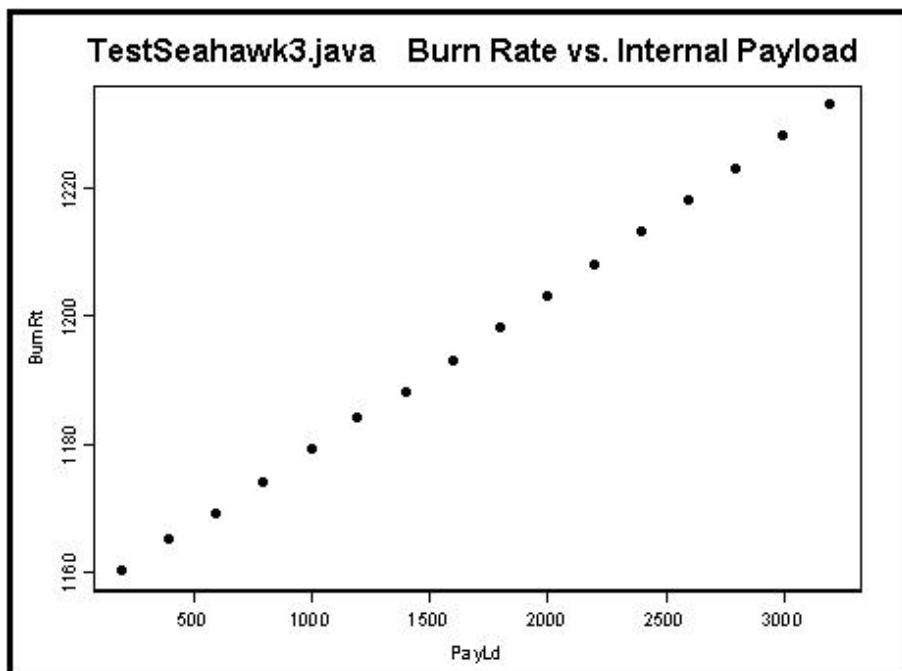


Figure 40. **TestSeahawk3** Fuel Consumption Rate Versus Internal Payload

TestSeahawk4 allows internal fuel carried by each configuration of the HH-60H to equal the maximum allowed based on internal payload weight up to the maximum internal fuel capacity. Figure 40 is a plot of fuel consumption rate versus radius of action

for this scenario. Again the relationship shows that the limiting (maximum) internal fuel capacity (4,012 lbs) causes the maximum radius of action to drop off significantly for the last ten configurations – again demonstrating an apparently linear relationship.

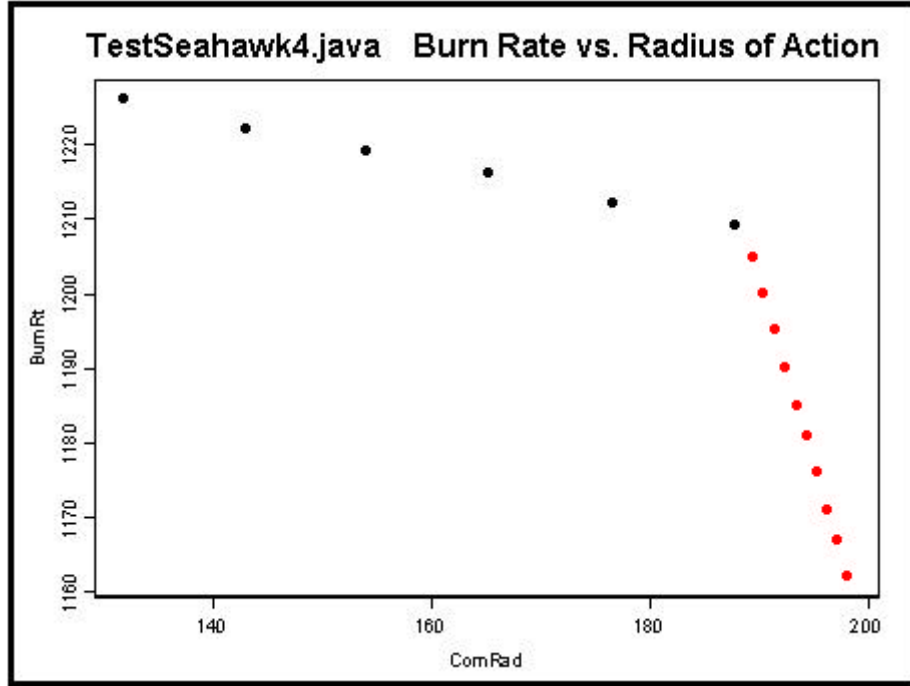


Figure 41. **TestSeahawk4** Fuel Consumption Rate Versus Radius of Action

Similarly, as for **TestSeahawk2**, we introduce a binary indicator variable to the model for **TestSeahawk4**. Again, a binary variable, *MaxFuel*, is added to the regression model such that *MaxFuel* equals one if the configuration launches with maximum internal fuel (4,012 lbs) and zero otherwise. The regression model is:

$$BurnRt = \beta_0 + \beta_1 * ComRad + \beta_2 * MaxFuel + \beta_3 * ComRad * MaxFuel$$

where: *ComRad* = radius of action (nautical miles)

MaxFuel = binary variable indicating maximum internal fuel

β_0 = regression model intercept

β_1, β_2 and β_3 = regression model coefficients

Regression Statistics	
Multiple R	0.999917003
R Square	0.999834013
Adjusted R Square	0.999792517
Standard Error	0.298276564
Observations	16

ANOVA					
	df	SS	MS	F	Significance F
Regression	3	6430.932373	2143.644124	24094.30609	6.13289E-23
Residual	12	1.067626907	0.088968909		
Total	15	6432			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept	1265.377022	1.023372046	1236.478002	5.27404E-32	1263.147286	1267.606758
ComRad	-0.300633187	0.006358251	-47.28237298	5.23667E-15	-0.314486625	-0.286779749
MaxFuel	868.972833	6.652889496	130.6158525	2.72112E-20	854.4774322	883.4682337
ComRad:MaxFuel	-4.603970819	0.034485064	-133.5062273	2.09292E-20	-4.679107318	-4.52883432

Table 14. **TestSeahawk4** Regression Output with Indicator Variable

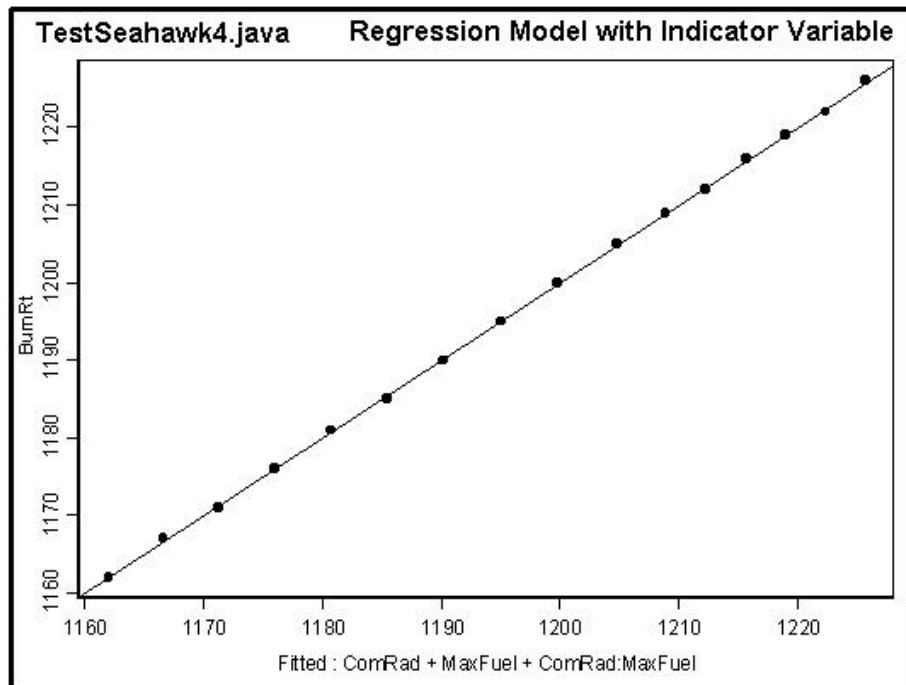


Figure 42. **TestSeahawk4** Actual Versus Fitted Values

The regression output shows the strong relationship between fuel consumption rate and radius of action, the binary variable determining maximum internal fuel and their interaction. In fact, the fit is nearly perfect - leading one to suspect that the residuals are interpolation errors from the original NATOPS performance charts.

V. CONCLUSIONS AND RECOMMENDATIONS

Mission radius and endurance are frequent KPPs on Operational Requirements Documents for military aircraft acquisitions. In fact, Figure 3 from Chapter I of this thesis registers mission radius as a KPP for every recent military aircraft acquisition program listed. Mission radius for a particular aircraft consists of that aircraft carrying a specified payload a specified distance (combat radius). Endurance for a particular aircraft consists of that aircraft carrying a specified payload for specified period of time at a specified altitude (on-station endurance).

This thesis examines seven different aircraft (AH-1W Cobra, AH-64A Apache, CH-53E Super Stallion, F-14A Tomcat, F/A-18 Hornet, HH-60H Seahawk and SH-60B Seahawk) all flying notional mission profiles meeting the requirements of MIL-P-2203 in which relationships between fuel efficiency and the factors combat radius, payload and on-station endurance are scrutinized. The resulting data analysis demonstrates that these relationships are strong. Consequently, any requirement to include fuel efficiency as a KPP is simply a duplication of effort.

The use of KPPs in an Operational Requirements Document should be kept to a minimum. The CJCS Instruction 3170.01B recommends no more than eight KPPs for any single acquisition program. The addition of a fuel efficiency KPP will not help the decision-maker determine a better alternative between competing airframes. Clearly, for a given fuel capacity, the airframe with the most fuel-efficient engine(s) is determined by the factors mission radius and/or endurance. The analysis in Chapter IV of this thesis proves the existence of these relationships and that these relationships are strong.

A. SURROGATES FOR FUEL EFFICIENCY

The issue of surrogates for fuel efficiency such as combat radius, payload and endurance are considered by the Defense Science Board report "More Capable Warfighting Through Reduced Fuel Burden" [DSB 2001], however they conclude that these are "weak" substitutes. In fact one of the authors stated while an F/A-18 could have a greater combat radius with a centerline tank, it is "much less fuel-efficient". Analysis using the Java model developed in this thesis does not support this premise.

Figure 41 is a graph of fuel consumption rate versus combat radius for sixty different configurations of the F/A-18C all flying the exact same mission profile described by **TestHornet2** (see Figure 28). For the first thirty configurations (config00 through config29) the centerline weapon station (station 5) is empty. The second thirty configurations (config30 through config59) are exactly the same as the first thirty configurations except that the centerline weapon station has a centerline 330-gallon fuel tank. As shown in Figure 41, essentially the combat radius shifts to the right. In fact, the overall average increase in combat radius is approximately 36% and the overall average increase in fuel consumption rate is less than 1%. One is likely to conclude that the increase in combat radius is well worth the insignificant increase in fuel consumption.

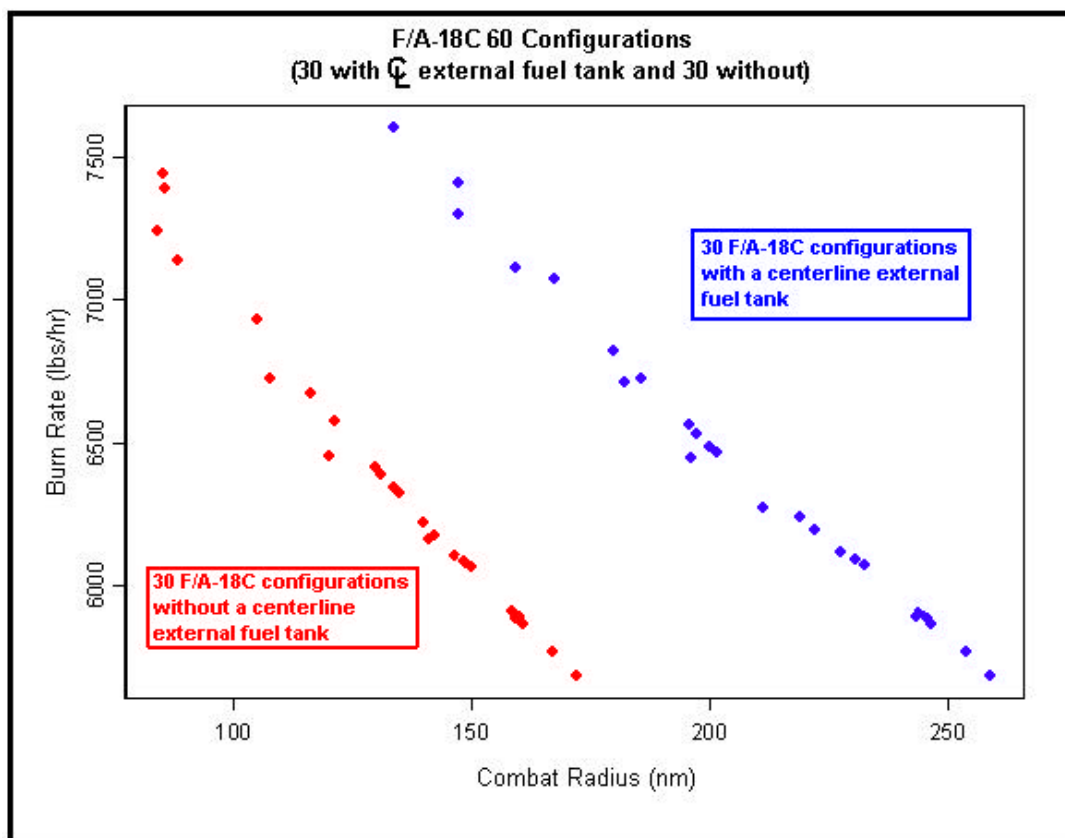


Figure 43. Effect of Additional Centerline Tank on F/A-18C Combat Radius

B. A MODEL FOR LOGISTICS PLANNING FACTORS

The Defense Science Board enumerated several shortcomings within DoD that do not properly account for the benefits of more fuel-efficient platforms. To their credit, they make many excellent recommendations. Perhaps their strongest point is the fact that

currently there are a lack of analytical tools within DoD that can be used to accurately reveal opportunities to improve fuel efficiency of weapons platforms. The Operations Research Department at the Naval Postgraduate School in Monterey, California has sponsored much research involving the flow of fuel in different regional CINC Areas of Responsibility (AORs) such as United States Central Command (CENTCOM) and United States Pacific Command (PACOM) among others. Recent thesis research has focused on the ability to meet the fuel demand for a Major Regional Contingency (MRC) [Iliakis 2002, Cha 2002, Devlin 2001, Noble 1999; et al]. There may be a need to develop tools for the logistics planning factors for the intra-theater fuel demand for a given MRC or Operational Plan. In other words - we may be able to use network analysis to find weaknesses within a given fuel distribution in-theater for a given demand, but is there a way to know what peak demand might look like? It is hoped that the methodology employed in the development of the Java model used in this thesis (**Aircraft**) might prove to be useful in the development of logistics planning factors for the development of just such demand levels.

Furthermore the **Aircraft** Java model could also be used in the progress of current and future DoD acquisitions and doctrine. The application of a similar model to current developmental warfare doctrine such as the U.S. Marine Corps' "Ship To Objective Maneuver" (STOM) could be useful in developing logistics planning factors for programs like the MV-22 Osprey and the Joint Strike Fighter, to name a few.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. STANDARD ATMOSPHERE

ICAO Standard Atmospheric Properties

Standard Sea Level Conditions:

Temperature: 15 Degrees Centigrade

Pressure: 29.921 inches Hg

Density: 0.0023769 slugs / ft³

Speed of Sound: 661.7 knots

Altitude (feet)	Density Ratio (σ)	Temperature (°C)	Speed of Sound (knots)
0	1.0000	15.00	661.7
1,000	0.9711	13.01	659.5
2,000	0.9428	11.05	657.2
3,000	0.9151	9.07	654.9
4,000	0.8881	7.08	652.6
5,000	0.8617	5.09	650.3
6,000	0.8359	2.99	647.9
7,000	0.8109	1.15	645.6
8,000	0.7860	-0.84	643.3
9,000	0.7620	-2.83	640.9
10,000	0.7385	-4.79	638.6
11,000	0.7157	-6.77	636.2
12,000	0.6933	-8.76	633.9
13,000	0.6715	-10.72	631.5
14,000	0.6502	-12.71	629.0
15,000	0.6295	-14.69	626.0
16,000	0.6092	-16.65	624.2
17,000	0.5895	-18.64	621.8
18,000	0.5702	-20.60	619.4
19,000	0.5514	-22.58	617.0
20,000	0.5332	-24.57	614.6
21,000	0.5153	-26.53	612.1
22,000	0.4980	-28.52	609.6
23,000	0.4811	-30.50	607.1
24,000	0.4646	-32.46	604.6
25,000	0.4486	-35.89	602.1
26,000	0.4330	-36.41	599.6
27,000	0.4178	-38.40	597.1
28,000	0.4030	-40.38	594.6
29,000	0.3887	-42.34	592.1
30,000	0.3747	-44.33	589.5
31,000	0.3611	-46.29	586.9
32,000	0.3479	-48.27	584.4

Altitude (feet)	Density Ratio (σ)	Temperature (°C)	Speed of Sound (kts)
33,000	0.3351	-50.23	581.8
34,000	0.3227	-52.22	579.2
35,000	0.3106	-54.18	576.6
36,000	0.2988	-56.16	574.0
36,089	0.2976	-56.45	573.7
37,000	0.2852	-56.45	573.7
38,000	0.2719	-56.45	573.7
39,000	0.2592	-56.45	573.7
40,000	0.2471	-56.45	573.7
41,000	0.2355	-56.45	573.7
42,000	0.2245	-56.45	573.7
43,000	0.2140	-56.45	573.7
44,000	0.2040	-56.45	573.7
45,000	0.1945	-56.45	573.7
46,000	0.1856	-56.45	573.7
47,000	0.1767	-56.45	573.7
48,000	0.1685	-56.45	573.7
49,000	0.1606	-56.45	573.7
50,000	0.1531	-56.45	573.7
52,000	0.1391	-56.45	573.7
54,000	0.1264	-56.45	573.7
56,000	0.1149	-56.45	573.7
58,000	0.1044	-56.45	573.7
60,000	0.0949	-56.45	573.7
62,000	0.0863	-56.45	573.7
64,000	0.0784	-56.45	573.7
66,000	0.0713	-56.45	573.7
68,000	0.0648	-56.45	573.7
70,000	0.0589	-56.45	573.7
72,000	0.0535	-56.45	573.7
74,000	0.0486	-56.45	573.7
76,000	0.0442	-56.45	573.7
78,000	0.0402	-56.45	573.7
80,000	0.0365	-56.45	573.7

APPENDIX B. AIRCRAFT CLASS JAVADOC CONSTRUCTOR AND METHOD SUMMARIES

A. INTERFACE AIRCRAFT

public interface **Aircraft**

An interface for use in aircraft fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for aircraft in various mission profiles.

Method Summary	
double[]	<u>climb</u> (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	<u>cruise</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	<u>descend</u> (double descendTo) This is a void method which simply changes the aircraft altitude.
double	<u>getClimbDist</u> (double alt) Returns the distance traveled over ground during a climb to the specified altitude (alt).
double	<u>getClimbFuel</u> (double alt) Returns the fuel consumed during a climb to the specified altitude (alt).
double	<u>getClimbTime</u> (double alt) Returns the time elapsed during a climb to the specified altitude (alt).
double	<u>getCurrentAlt</u> () Returns altitude (the current altitude of the aircraft).
double	<u>getCurrentGS</u> () Returns groundSpeed (the current speed over ground of the aircraft - equal to True Air Speed).
double	<u>getCurrentGW</u> () Returns grossWt (the current gross weight of the aircraft).
double[]	<u>getLevelFltPerf</u> (double alt) Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed

	(Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmcp), and Maximum Continuous Power Fuel Flow (FFmcp).
double	<u>getSurfTemp</u> () Returns surfTemp (surface temperature at takeoff airfield) instance variable.
double	<u>interpolate</u> (double lower, double percent, double upper) Returns the interpolated value between the lower and upper doubles.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>setGrossWt</u> (double inputWt) Sets aircraft grossWt (gross weight of aircraft) instance variable.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument

B. CLASS AH1W

All Implemented Interfaces:

Aircraft

```
public class AH1W
extends java.lang.Object
implements Aircraft
```

A class for use in AH-1W Cobra Helicopter fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the AH-1W Cobra Helicopter in various mission profiles. The AH-1W Cobra is an all-weather attack helicopter capable of carrying multiple payload combinations of air-to-ground and air-to-air weapons.

Constructor Summary

AH1W(double freeAirTemp, double takeOffWeight, double fieldElev, double drag)

Constructor arguments consist of free air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), field elevation at takeoff (measured from sea level), and total drag effect (computed per Paragraph 28.2 of NAVAIR 01-H1AAC-1 AH-1W NATOPS Manual).

Method Summary

double[]	climb (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	cruise (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	descend (double descendTo) This is a void method which simply changes the aircraft altitude.
double	getClimbDist (double alt) Returns the corresponding distance-travelled-in-climb for a given index and surface temperature from the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1) Figure 27-1 (top).
double	getClimbFuel (double alt) Returns the corresponding fuel-used-in-climb for a given index and

	surface temperature from the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1) Figure 27-1 (top).
double	<u>getClimbTime</u> (double alt) Returns the corresponding time-to-climb for a given index and surface temperature from the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1) Figure 27-1 (top).
double	<u>getClimbX</u> (double alt) Returns the corresponding index for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1) Figure 27-1 (top).
double	<u>getCurrentAlt</u> () Returns the AH-1W helicopter current altitude.
double	<u>getCurrentGS</u> () Returns the AH-1W helicopter current speed over ground.
double	<u>getCurrentGW</u> () Returns AH-1W helicopter gross weight.
double[]	<u>getEightKMnus10ToPlus10Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -10 deg C and +10 deg C at 8,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getEightKMnus30ToMnus10Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -30 deg C and -10 deg C at 8,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getEightKPlus10ToPlus35Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +10 deg C and +35 deg C at 8,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getFourKfMnus05ToPlus20Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -5 deg C and +20 deg C at 4,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).

double[]	<u>getFourKfMnus30ToMnus05Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -30 deg C and -5 deg C at 4,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getFourKfPlus20ToPlus45Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +20 deg C and +45 deg C at 4,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getLevelFltPerf</u> (double alt) Returns the array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for the aircraft interpolated for the current outside air temperature and altitude and gross weight of aircraft.
double[]	<u>getSeaLvlMnus25ToPlus05Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -25 deg C and +5 deg C at sea level from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getSeaLvlPlus05ToPlus25Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +5 deg C and +25 deg C at sea level from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getSeaLvlPlus25ToPlus50Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +25 deg C and +50 deg C at sea level from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getSixKftMnus10ToPlus15Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum

	range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -10 deg C and +15 deg C at 6,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getSixKftMnus30ToMnus10Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -30 deg C and -10 deg C at 6,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getSixKftPlus15ToPlus35Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +15 deg C and +35 deg C at 6,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double	<u>getSurfTemp</u> () Returns surface temperature at takeoff airfield.
int	<u>getSurfTempRow</u> () Returns the matrix row corresponding to the temperature at altitude for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1) Figure 27-1 (top).
double[]	<u>getTenKftMnus15ToPlus05Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -15 deg C and +5 deg C at 10,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getTenKftMnus35ToMnus15Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -35 deg C and -15 deg C at 10,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getTenKftPlus05ToPlus30Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +5 deg C and +30 deg C at 10,000 ft MSL from Figure

	28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getTwoKftMnus30ToPlus00Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between -30 deg C and +0 deg C at 2,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getTwoKftPlus00ToPlus20Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +0 deg C and +20 deg C at 2,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double[]	<u>getTwoKftPlus20ToPlus45Parameters</u> (double alt, double oAT) Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperatures between +20 deg C and +45 deg C at 2,000 ft MSL from Figure 28-1 of the AH-1W Cobra NATOPS Manual (NAVAIR 01-H1AAC-1).
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>setDragEffect</u> (double dragTotal) Sets aircraft drag effect for the AH-1W helicopter.
void	<u>setGrossWt</u> (double inputWt) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

C. CLASS AH64A

All Implemented Interfaces:

Aircraft

```
public class AH64A
extends java.lang.Object
implements Aircraft
```

A class for use in AH-64A Apache Helicopter fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the AH-64A Apache Helicopter in various mission profiles. The AH-64A Apache is an all-weather attack helicopter used by the U.S. Army, capable of carrying multiple payload combinations of air-to-ground and air-to-air weapons.

Constructor Summary

AH64A(double freeAirTemp, double takeOffWeight, double fieldElev, double drag)

Constructor arguments consist of free air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), field elevation at takeoff airfield (measured from sea level), and total drag effect (computed per Figure 7-60 of TM 55-1520-238-10 Technical Manual).

Method Summary

double[]	<u>climb</u> (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	<u>cruise</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	<u>descend</u> (double descendTo) This is a void method which simply changes the aircraft altitude.
double	<u>getClimbDist</u> (double alt) Returns the corresponding distance-travelled-in-climb for a given index and surface temperature from the AH-64A Apache Technical Manual (TM 55-1520-238-10) Figure 7-63 (top).
double	<u>getClimbFuel</u> (double alt) Returns the corresponding fuel-used-in-climb for a given index and

	surface temperature from the AH-64A Apache Technical Manual (TM 55-1520-238-10) Figure 7-63 (top).
double	<u>getClimbTime</u> (double alt) Returns the corresponding time-to-climb for a given index and surface temperature from the AH-64A Apache Technical Manual (TM 55-1520-238-10) Figure 7-63 (top).
double	<u>getClimbX</u> (double alt) Returns the corresponding index for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the AH-64A Apache Technical Manual (TM 55-1520-238-10) Figure 7-63 (top).
double	<u>getCurrentAlt</u> () Returns the AH-64A helicopter current altitude.
double	<u>getCurrentGS</u> () Returns the AH-64A helicopter current speed over ground.
double	<u>getCurrentGW</u> () Returns AH-64A helicopter gross weight.
double[]	<u>getEightKMnus10Parameters</u> () Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at 8,000 ft from Figure 7-36 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKMnus20Parameters</u> () Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at 8,000 ft from Figure 7-35 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKMnus30Parameters</u> () Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at 8,000 ft from Figure 7-35 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKMnus40Parameters</u> () Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at 8,000 ft from Figure 7-34 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).

double[]	<u>getEightKMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at 8,000 ft from Figure 7-34 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at 8,000 ft from Figure 7-36 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at 8,000 ft from Figure 7-37 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +20 deg C at 8,000 ft from Figure 7-37 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at 8,000 ft from Figure 7-38 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getEightKPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at 8,000 ft from Figure 7-38 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfMnus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum

	range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at 4,000 ft from Figure 7-24 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfMnus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at 4,000 ft from Figure 7-24 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfMnus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at 4,000 ft from Figure 7-23 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfMnus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at 4,000 ft from Figure 7-23 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at 4,000 ft from Figure 7-22 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at 4,000 ft from Figure 7-25 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at 4,000 ft from Figure 7-25 of the AH-64A Apache

	Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +20 deg C at 4,000 ft from Figure 7-26 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at 4,000 ft from Figure 7-26 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at 4,000 ft from Figure 7-27 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getFourKfPlus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +50 deg C at 4,000 ft from Figure 7-27 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getLevelFltPerf()</u> (double alt) Returns the array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for the aircraft interpolated for the current outside air temperature and altitude and gross weight of aircraft.
double[]	<u>getSeaLvlMnus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at sea level from Figure 7-10 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlMnus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum

	endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at sea level from Figure 7-10 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlMnus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at sea level from Figure 7-9 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlMnus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at sea level from Figure 7-9 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at sea level from Figure 7-8 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at sea level from Figure 7-11 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at sea level from Figure 7-11 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air

	temperature +20 deg C at sea level from Figure 7-12 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at sea level from Figure 7-12 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at sea level from Figure 7-13 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSeaLvlPlus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +50 deg C at sea level from Figure 7-13 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftMnus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at 6,000 ft from Figure 7-31 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftMnus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at 6,000 ft from Figure 7-30 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftMnus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at 6,000 ft from Figure 7-30 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftMnus40Parameters()</u>

	Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at 6,000 ft from Figure 7-29 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at 6,000 ft from Figure 7-29 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at 6,000 ft from Figure 7-31 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at 6,000 ft from Figure 7-32 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +20 deg C at 6,000 ft from Figure 7-32 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at 6,000 ft from Figure 7-33 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getSixKftPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and

	maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at 6,000 ft from Figure 7-33 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double	<u>getSurfTemp()</u> Returns surface temperature at takeoff airfield.
int	<u>getSurfTempRow()</u> Returns the matrix row corresponding to the temperature at altitude for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the AH-64A Apache Technical Manual (TM 55-1520-238-10) Figure 7-63 (top).
double[]	<u>getTenKftMnus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at 10,000 ft from Figure 7-41 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftMnus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at 10,000 ft from Figure 7-40 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftMnus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at 10,000 ft from Figure 7-40 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftMnus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at 10,000 ft from Figure 7-39 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at 10,000 ft from Figure 7-39 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).

double[]	<u>getTenKftPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at 10,000 ft from Figure 7-41 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at 10,000 ft from Figure 7-42 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +20 deg C at 10,000 ft from Figure 7-42 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at 10,000 ft from Figure 7-43 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTenKftPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at 10,000 ft from Figure 7-44 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftMnus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -10 deg C at 2,000 ft from Figure 7-17 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftMnus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum

	range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -20 deg C at 2,000 ft from Figure 7-17 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftMnus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -30 deg C at 2,000 ft from Figure 7-16 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftMnus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -40 deg C at 2,000 ft from Figure 7-16 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftMnus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature -50 deg C at 2,000 ft from Figure 7-15 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus00Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature 0 deg C at 2,000 ft from Figure 7-18 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus10Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +10 deg C at 2,000 ft from Figure 7-18 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus20Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +20 deg C at 2,000 ft from Figure 7-19 of the AH-64A Apache

	Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus30Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +30 deg C at 2,000 ft from Figure 7-19 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus40Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +40 deg C at 2,000 ft from Figure 7-20 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double[]	<u>getTwoKftPlus50Parameters()</u> Returns an array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for outside air temperature +50 deg C at 2,000 ft from Figure 7-20 of the AH-64A Apache Technical Manual (TM 55-1520-238-10).
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>setDragFactor</u> (double dragNum) Sets aircraft drag factor for the AH-64A helicopter.
void	<u>setGrossWt</u> (double weight) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

D. CLASS CH53E

All Implemented Interfaces:

Aircraft

```
public class CH53E
extends java.lang.Object
implements Aircraft
```

A class for use in CH-53E Super Stallion Helicopter fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the CH-53E Super Stallion Helicopter in various mission profiles. The CH-53E Super Stallion is a heavy lift helicopter used by the U.S. Marine Corps, capable of carrying various external cargo payloads.

Constructor Summary

[**CH53E**](#)(double outsideAirTemp, double takeOffWeight, double fieldElev)

Constructor arguments consist of free air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), and field elevation at takeoff airfield (measured from sea level).

Method Summary

void	ascend (double ascendTo) This is a void method which simply changes the aircraft altitude.
double[]	climb (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	cruise (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	descend (double descendTo) This is a void method which simply changes the aircraft altitude.
double	get3EngFFme (double alt) Returns the corresponding maximum continuous power fuel flow value based outside aircraft gross weight and altitude from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 23-1 (assumes Standard Atmosphere).

double	<u>get3EngMnus40SRmr</u> (double alt) Returns the corresponding specific range value based on aircraft gross weight and altitude for standard atmospheric conditions minus 40 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-3.
double	<u>get3EngMnus40Vmr</u> (double alt) Returns the corresponding maximum range airspeed value based on aircraft gross weight and altitude for standard atmospheric conditions minus 40 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-3.
double	<u>get3EngPlus40SRmr</u> (double alt) Returns the corresponding specific range value based on aircraft gross weight and altitude for standard atmospheric conditions plus 40 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-1.
double	<u>get3EngPlus40Vmr</u> (double alt) Returns the corresponding maximum range airspeed value based on aircraft gross weight and altitude for standard atmospheric conditions plus 40 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-1.
double	<u>get3EngSRmcp</u> (double alt) Returns the corresponding maximum continuous power specific range value based outside aircraft gross weight and altitude from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-4 (assumes Standard Atmosphere).
double	<u>get3EngSRmr</u> (double alt) Returns the corresponding specific range value based on outside air temperature from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-1, 22-2 & 22-3.
double	<u>get3EngStdDaySRmr</u> (double alt) Returns the corresponding specific range value based on aircraft gross weight and altitude for standard atmospheric conditions from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-2.
double	<u>get3EngStdDayVmr</u> (double alt) Returns the corresponding maximum range airspeed value based on aircraft gross weight and altitude for standard atmospheric conditions from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-2.
double	<u>get3EngVmcp</u> (double alt) Returns the corresponding maximum continuous power airspeed value based outside aircraft gross weight and altitude from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-4 (assumes Standard Atmosphere).
double	<u>get3EngVme</u> (double alt)

	Returns the corresponding maximum endurance airspeed value based outside aircraft gross weight and altitude from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 23-1 (assumes Standard Atmosphere).
double	<u>get3EngVmr</u> (double alt) Returns the corresponding maximum range airspeed value based outside air temperature from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 22-1, 22-2 & 22-3.
double	<u>getClimbDist</u> (double alt) Returns the corresponding distance-travelled-in-climb based on outside air temperature from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 21-1 and 21-2.
double	<u>getClimbFuel</u> (double alt) Returns the corresponding fuel-used-in-climb based on outside air temperature from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 21-1 and 21-2.
double	<u>getClimbTime</u> (double alt) Returns the corresponding time-to-climb based on outside air temperature from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figures 21-1 and 21-2.
double	<u>getColdClimbDistX</u> (double alt) Returns the corresponding index for use in the distance-travelled-in-climb chart when outside air temperature is less than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getColdClimbFuelX</u> (double alt) Returns the corresponding index for use in the fuel-used-in-climb chart when outside air temperature is less than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getColdClimbTimeX</u> (double alt) Returns the corresponding index for use in the time-to-climb chart when outside air temperature is less than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getColdDayClimbDist</u> (double coldx) Returns the corresponding distance-travelled-in-climb for a given index and surface temperature (less than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getColdDayClimbFuel</u> (double coldx) Returns the corresponding fuel-used-in-climb for a given index and surface temperature (less than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getColdDayClimbTime</u> (double coldx) Returns the corresponding time-to-climb for a given index and surface temperature (less than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.

int	<u>getColdSurfTempCol()</u> Returns the matrix column corresponding to the temperature at takeoff for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getCurrentAlt()</u> Returns the CH-53E helicopter current altitude.
double	<u>getCurrentGS()</u> Returns the CH-53E helicopter current speed over ground.
double	<u>getCurrentGW()</u> Returns CH-53E helicopter gross weight.
double	<u>getHoverFuelFlow(double alt)</u> Returns hover performance fuel flow for the CH-53E "out-of-ground-effect" hover evolution from CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 23-5.
double[]	<u>getLevelFltPerf(double alt)</u> Returns the array of maximum endurance airspeed (Vme), maximum endurance fuel flow (FFme), maximum range airspeed (Vmr), maximum range fuel flow (FFmr), maximum continuous power airspeed (Vmcp), and maximum continuous power fuel flow (FFmcp) values for the aircraft interpolated for the current outside air temperature and altitude and gross weight of aircraft.
double	<u>getSigma(double alt)</u> Returns the corresponding value of sigma (ratio of Speed of Sound at Altitude to Speed of Sound at Sea Level) for a constant altitude from the AIRCRAFT ENGINE DESIGN (Mattingly, Jack K.
double	<u>getSurfTemp()</u> Returns surface temperature at takeoff airfield.
double[]	<u>getVERTREPerf(double alt, double dragArea)</u> Returns an array of external cargo airspeed (Vext) and external cargo fuel flow (FFext) for the CH-53E carrying an external payload.
double	<u>getWarmClimbDistX(double alt)</u> Returns the corresponding index for use in the distance-travelled-in-climb chart when outside air temperature is greater than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2.
double	<u>getWarmClimbFuelX(double alt)</u> Returns the corresponding index for use in the fuel-used-in-climb chart when outside air temperature is greater than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2.
double	<u>getWarmClimbTimeX(double alt)</u> Returns the corresponding index for use in the time-to-climb chart when outside air temperature is greater than 0 deg C from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-1.
double	<u>getWarmDayClimbDist(double warmX)</u>

	Returns the corresponding distance-travelled-in-climb for a given index and surface temperature (greater than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2.
double	<u>getWarmDayClimbFuel</u> (double warmX) Returns the corresponding fuel-used-in-climb for a given index and surface temperature (greater than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2.
double	<u>getWarmDayClimbTime</u> (double warmX) Returns the corresponding time-to-climb for a given index and surface temperature (greater than 0 deg C) from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2.
int	<u>getWarmSurfTempCol</u> () Returns the matrix column corresponding to the temperature at takeoff for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the CH-53E NATOPS Manual (NAVAIR A1-H53BE-NFM-000) Figure 21-2 for warm day conditions (outside air temperature greater than 0 deg C).
double[]	<u>hover</u> (double time) Returns the distance traveled over ground (always zero), time elapsed, and fuel consumed during a hover flight evolution where time elapsed is the method argument.
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>pickUpLoad</u> (double cargo) This is a void method which simply increases the aircraft gross weight by the weight of the external cargo load.
void	<u>releaseLoad</u> (double cargo) This is a void method which simply decreases the aircraft gross weight by the weight of the external cargo load.
void	<u>setGrossWt</u> (double weight) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.
double[]	<u>vertrep</u> (double distance, double dragA)

	Returns the distance traveled over ground, time elapsed, and fuel consumed during an external cargo flight evolution where distance travelled and external cargo drag area are the method arguments.
--	--

E. CLASS F14AStd

All Implemented Interfaces:
Aircraft

```
public class F14AStd
extends java.lang.Object
implements Aircraft
```

A class for use in F-14A Tomcat fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the F-14A Tomcat in various mission profiles. The F-14A Tomcat is an fighter (and attack) aircraft used by the U.S. Navy, capable of carrying multiple combinations of weapon payloads.

Constructor Summary

F14AStd(double outsideAirTemp, double takeOffWeight, double fieldElev, double drag)

Constructor arguments consist of outside air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), field elevation at takeoff airfield (measured from sea level), and total drag index (computed from NWP 3-22.5-F14A/B/D, Volume III Figure 1-8).

Method Summary

double[]	climb (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double	convertCASToTAS (double alt, double cAS) Returns the true airspeed converted from calibrated airspeed.
double[]	cruise (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	descend (double descendTo) This is a void method which simply changes the aircraft altitude.
double	getClimbDist (double alt) Returns the corresponding distance-travelled-in-climb based on gross

	weight, altitude, and drag index from the F-14A NATOPS Performance Charts (NAVAIR 01-F14AAA-1.1) Figure 3-1.
double	<u>getClimbFuel</u> (double alt) Returns the corresponding fuel-used-in-climb based on gross weight, altitude, and drag index from the F-14A NATOPS Performance Charts (NAVAIR 01-F14AAA-1.1) Figure 3-1.
double	<u>getClimbTime</u> (double alt) Returns the corresponding time-to-climb based on gross weight, altitude, and drag index from the F-14A NATOPS Performance Charts (NAVAIR 01-F14AAA-1.1) Figure 3-1.
double[]	<u>getCruiseParameters</u> (double alt) Returns an array of maximum range airspeed and maximum range fuel flow (Vmr & FFmr).
double	<u>getCurrentAlt</u> () Returns the F-14A current altitude.
double	<u>getCurrentGS</u> () Returns the F-14A current speed over ground.
double	<u>getCurrentGW</u> () Returns F-14A current gross weight.
double[]	<u>getEnduranceParameters</u> (double alt) Returns an array of maximum endurance airspeed and maximum endurance fuel flow (Vme & FFme).
double[]	<u>getLevelFltPerf</u> (double alt) Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed (Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmcp), and Maximum Continuous Power Fuel Flow (FFmcp).
double[]	<u>getMaxABparameters</u> (double alt) Returns an array of maximum afterburner airspeed and maximum afterburner fuel flow (Vmr & FFmr).
double[]	<u>getMilPwrParameters</u> (double alt) Returns an array of maximum continuous power airspeed and maximum continuous power fuel flow (Vmr & FFmr).
double	<u>getSpdSoundAtAltitude</u> (double altitude) Returns the speed of sound at altitude for use in converting mach number to true airspeed.
double	<u>getSurfTemp</u> () Returns surface temperature at takeoff airfield.
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel

	consumed during a loiter flight evolution.
void	<u>setGrossWt</u> (double weight) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.
double[]	<u>supersonicDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a supersonic flight evolution where distance travelled is the method argument.
double[]	<u>supersonicTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a supersonic flight evolution where time elapsed is the method argument.

F. CLASS FA18CStd

All Implemented Interfaces:

Aircraft

```
public class FA18CStd
extends java.lang.Object
implements Aircraft
```

A class for use in F/A-18C Hornet fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the F/A-18C Hornet in various mission profiles. The F/A-18C Hornet is a fighter/attack aircraft used by the U.S. Navy, capable of carrying multiple combinations of weapon payloads.

Constructor Summary

```
FA18CStd(double outsideAirTemp, double takeOffWeight,
double fieldElev, double bSDrag, double intDragCd)
```

Constructor arguments consist of outside air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), field elevation at takeoff airfield (measured from sea level), basic stores drag index (computed from F/A-18C NATOPS

Performance Charts Figure 11-2) and interference drag code (computed from F/A-18C NATOPS Performance Charts Figure 11-3 (NAVAIR A1-F18AC-NFM-200)) .

Method Summary

double[]	<u>climb</u> (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	<u>cruise</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	<u>descend</u> (double descendTo) This is a void method which simply changes the aircraft altitude.
double	<u>getBasicStDrag</u> () Returns F/A-18C basic stores drag index.
double	<u>getClimbDist</u> (double alt) Returns the corresponding distance-travelled-in-climb based on gross weight, altitude, and drag index from the F/A-18C NATOPS Performance Charts (NAVAIR A1-F18AC-NFM-200) Figure 11-23.
double	<u>getClimbFuel</u> (double alt) Returns the corresponding fuel-used-in-climb based on gross weight, altitude, and drag index from the F/A-18C NATOPS Performance Charts (NAVAIR A1-F18AC-NFM-200) Figure 11-23.
double	<u>getClimbTime</u> (double alt) Returns the corresponding time-to-climb based on gross weight, altitude, and drag index from the F/A-18C NATOPS Performance Charts (NAVAIR A1-F18AC-NFM-200) Figure 11-23.
double	<u>getCurrentAlt</u> () Returns F/A-18C current altitude.
double	<u>getCurrentGS</u> () Returns F/A-18C current speed over ground.
double	<u>getCurrentGW</u> () Returns F/A-18C current gross weight.
double	<u>getDragIndex</u> (double machNumber) Returns F/A-18C total drag index - sum of the basic stores drag index and interference drag index (determined by the interference drag code in conjunction with Figure 11-4 from F/A-18C NATOPS Performance Charts (NAVAIR A1-F18AC-NFM-200)).
double	<u>getIntDragCode</u> () Returns F/A-18C interference drag code.
double[]	<u>getLevelFltPerf</u> (double alt)

	Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed (Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmcp), and Maximum Continuous Power Fuel Flow (FFmcp).
double	<u>getSpdSoundAtAltitude</u> (double altitude) Returns the speed of sound at altitude for use in converting mach number to true airspeed.
double	<u>getSurfTemp</u> () Returns surface temperature at takeoff airfield.
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>setDragIndex</u> (double machNo) Sets aircraft drag index - changes as airspeed changes for some payload combinations.
void	<u>setGrossWt</u> (double gW) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

G. CLASS HH60H

All Implemented Interfaces:

Aircraft

```
public class HH60H
extends java.lang.Object
implements Aircraft
```

A class for use in HH-60H Helicopter fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the HH-60H Seahawk Helicopter in various mission profiles. The HH-60H Helicopter is a versatile

platform used by the U.S. Navy for Combat Search and Rescue, Naval Special Warfare and utility missions.

Constructor Summary

[HH60H](#)(double outsideAirTemp, double takeOffWeight, double fieldElev)

Constructor arguments consist of outside air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), and field elevation at takeoff airfield (measured from sea level).

Method Summary

double[]	climb (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	cruise (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	descend (double descendTo) This is a void method which simply changes the aircraft altitude.
double	getClimbDist (double alt) Returns the corresponding distance-travelled-in-climb for a given index and surface temperature from the HH-60H Seahawk NATOPS Manual (NAVAIR A1-H60CA-NFM-000) Figure 24-2.
double	getClimbFuel (double alt) Returns the corresponding fuel-used-in-climb for a given index and surface temperature from the HH-60H Seahawk NATOPS Manual (NAVAIR A1-H60CA-NFM-000) Figure 24-2.
double	getClimbTime (double alt) Returns the corresponding time-to-climb for a given index and surface temperature from the HH-60H Seahawk NATOPS Manual (NAVAIR A1-H60CA-NFM-000) Figure 24-2.
double	getClimbX (double alt) Returns the corresponding index for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the HH-60H Seahawk NATOPS Manual (NAVAIR A1-H60CA-NFM-000) Figure 24-2 (top).
double	getCurrentAlt () Returns HH-60H current altitude.
double	getCurrentGS () Returns HH-60H current speed over ground.

double	<u>getCurrentGW()</u> Returns HH-60H current gross weight.
double[]	<u>getLevelFltPerf</u> (double alt) Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed (Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmcp), and Maximum Continuous Power Fuel Flow (FFmcp).
double	<u>getSurfTemp</u> () Returns surface temperature at takeoff airfield.
int	<u>getSurfTempRow</u> () Returns the matrix row corresponding to the temperature at altitude for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the HH-60H Seahawk NATOPS Manual (NAVAIR A1-H60CA-NFM-000) Figure 24-2 (top).
double	<u>interpolate</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>setGrossWt</u> (double weight) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

H. CLASS SH60B

All Implemented Interfaces:

Aircraft

```
public class SH60B
extends java.lang.Object
implements Aircraft
```

A class for use in SH-60B Seahawk Helicopter fuel consumption modeling.

The functionality is the ability to determine fuel consumption rates for the SH-60B Seahawk Helicopter in various mission profiles. The SH-60B Seahawk Helicopter is an all-purpose helicopter used by the U.S. Navy for Undersea Warfare, Anti-Ship Surveillance and Targeting and Search and Rescue.

Constructor Summary

SH60B(double outsideAirTemp, double takeOffWeight, double fieldElev)

Constructor arguments consist of outside air temperature at takeoff field (assumes Standard Atmospheric properties), total aircraft weight at takeoff (includes aircraft basic weight + crew weight + fuel weight + payload weight), and field elevation at takeoff airfield (measured from sea level).

Method Summary

double[]	climb (double climbTo) Returns the distance traveled over ground, time elapsed, and fuel consumed during a climb flight evolution.
double[]	cruise (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a cruise flight evolution.
void	descend (double descendTo) This is a void method which simply changes the aircraft altitude.
void	fireAGM114 () This is a void method which reduces the aircraft gross weight by 160 lbs.
void	fireAGM119B () This is a void method which reduces the aircraft gross weight by 860 lbs.
double	getClimbDist (double alt) Returns the corresponding distance-travelled-in-climb for a given index and surface temperature from the SH-60B Seahawk NATOPS Manual (NAVAIR A1-H60BB-NFM-000) Figure 24-1.
double	getClimbFuel (double alt) Returns the corresponding fuel-used-in-climb for a given index and surface temperature from the SH-60B Seahawk NATOPS Manual (NAVAIR A1-H60BB-NFM-000) Figure 24-1.
double	getClimbTime (double alt) Returns the corresponding time-to-climb for a given index and surface temperature from the SH-60B Seahawk NATOPS Manual (NAVAIR A1-H60BB-NFM-000) Figure 24-1.
double	getClimbX (double alt)

	Returns the corresponding index for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the SH-60B Seahawk NATOPS Manual (NAVAIR A1-H60BB-NFM-000) Figure 24-1 (top).
double	<u>getCurrentAlt()</u> Returns SH-60B current altitude.
double	<u>getCurrentGS()</u> Returns SH-60B current speed over ground.
double	<u>getCurrentGW()</u> Returns SH-60B current gross weight.
double[]	<u>getLevelFltPerf()</u> (double alt) Returns the array of Maximum Endurance Airspeed (Vme), Maximum Endurance Fuel Flow (FFme), Maximum Range Airspeed (Vmr), Maximum Range Fuel Flow (FFmr), Maximum Continuous Power Airspeed (Vmcp), and Maximum Continuous Power Fuel Flow (FFmcp).
double	<u>getSurfTemp()</u> Returns surface temperature at takeoff airfield.
int	<u>getSurfTempRow()</u> Returns the matrix row corresponding to the temperature at altitude for use in the time-to-climb, fuel-used-in-climb, and distance-travelled-in-climb charts from the SH-60B Seahawk NATOPS Manual (NAVAIR A1-H60BB-NFM-000) Figure 24-1 (top).
double	<u>interpolate()</u> (double lower, double percent, double upper) A general interpolation method used for reading various flight performance matrices within this class.
double[]	<u>loiter()</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a loiter flight evolution.
void	<u>releaseDICASS()</u> This is a void method which reduces the aircraft gross weight by 43 lbs.
void	<u>releaseDIFAR()</u> This is a void method which reduces the aircraft gross weight by 33 lbs.
void	<u>releaseMK46()</u> This is a void method which reduces the aircraft gross weight by 548 lbs.
void	<u>releaseMK50()</u> This is a void method which reduces the aircraft gross weight by 750 lbs.
void	<u>releaseVLAD()</u> This is a void method which reduces the aircraft gross weight by 33 lbs.

void	<u>setGrossWt</u> (double weight) Sets aircraft gross weight.
double[]	<u>speedyDist</u> (double distance) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where distance over ground is the method argument.
double[]	<u>speedyTime</u> (double time) Returns the distance traveled over ground, time elapsed, and fuel consumed during a speedy flight evolution where time elapsed is the method argument.

LIST OF REFERENCES

Chairman, Joints Chiefs of Staff, Requirements Generation System CJCSI 3170.01B. Washington DC: 2001.

Mattingly, Jack D., Heiser, William H., and Daley, Daniel H. Aircraft Engine Design. New York: American Institute of Aeronautics and Astronautics, Inc. 1987.

Johns Hopkins University Applied Physics Laboratory. Fuel Efficiency as a Key Performance Parameter in System Life Cycle Analyses of Aircraft Fleets. Draft Working Paper. 2001

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization Manual (NATOPS) AH-1W Super Cobra (NAVAIR 01-H1AAC-1). Patuxent River Naval Air Station: 2001.

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization Manual (NATOPS) CH-53E Super Stallion (NAVAIR A1-H53BE-NFM-000). Patuxent River Naval Air Station: 2001.

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization (NATOPS) Flight Performance Charts F/A-18C Hornet (NAVAIR A1-F18AC-NFM-200). Patuxent River Naval Air Station: 1999.

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization (NATOPS) Flight Performance Charts F-14A Tomcat (NAVAIR 01-F14AAA-1.1). Patuxent River Naval Air Station: 1998.

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization Manual (NATOPS) HH-60H Seahawk (NAVAIR A1-H60BB-NFM-000). Patuxent River Naval Air Station: 2001.

Naval Air Systems Command, Naval Air Training and Operating Procedures Standardization Manual (NATOPS) SH-60B Seahawk (NAVAIR A1-H60CA-NFM-000). Patuxent River Naval Air Station: 2002.

Naval Air Systems Command, Standard Aircraft Characteristics A-4C Skyhawk. Patuxent River Naval Air Station: 1970.

Naval Air Systems Command, Standard Aircraft Characteristics A-4E Skyhawk. Patuxent River Naval Air Station: 1972.

Naval Air Systems Command, Standard Aircraft Characteristics A-6A Intruder. Patuxent River Naval Air Station: 1960.

Naval Air Systems Command, Standard Aircraft Characteristics A-6E Intruder. Patuxent River Naval Air Station: 1971.

Naval Air Systems Command, Standard Aircraft Characteristics A-6E (TRAM) Intruder. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics A-7E Corsair. Patuxent River Naval Air Station: 1976.

Naval Air Systems Command, Standard Aircraft Characteristics AH-1W Super Cobra. Patuxent River Naval Air Station: 1988.

Naval Air Systems Command, Standard Aircraft Characteristics AV-8B Harrier. Patuxent River Naval Air Station: 1976.

Naval Air Systems Command, Standard Aircraft Characteristics C-2A Greyhound. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics C-9B Skytrain. Patuxent River Naval Air Station: 1972.

Naval Air Systems Command, Standard Aircraft Characteristics CH-46D Sea Knight. Patuxent River Naval Air Station: 1985.

Naval Air Systems Command, Standard Aircraft Characteristics CH-46E Sea Knight. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics CH-53E Super Stallion. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics E-2C Hawkeye. Patuxent River Naval Air Station: 1989.

Naval Air Systems Command, Standard Aircraft Characteristics E-6A TACAMO. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics EA-3B Skywarrior. Patuxent River Naval Air Station: 1959.

Naval Air Systems Command, Standard Aircraft Characteristics EA-6B Prowler. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics EC-130Q Hercules. Patuxent River Naval Air Station: 1976.

Naval Air Systems Command, Standard Aircraft Characteristics F-4S Phantom II. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics F-8C Crusader. Patuxent River Naval Air Station: 1959.

Naval Air Systems Command, Standard Aircraft Characteristics F-14D Tomcat. Patuxent River Naval Air Station: 1985.

Naval Air Systems Command, Standard Aircraft Characteristics F-16N Falcon. Patuxent River Naval Air Station: 1978.

Naval Air Systems Command, Standard Aircraft Characteristics F/A-18C Hornet. Patuxent River Naval Air Station: 1995.

Naval Air Systems Command, Standard Aircraft Characteristics F/A-18E Hornet. Patuxent River Naval Air Station: 1993.

Naval Air Systems Command, Standard Aircraft Characteristics HH-60H Seahawk. Patuxent River Naval Air Station: 1988.

Naval Air Systems Command, Standard Aircraft Characteristics KC-130F Hercules. Patuxent River Naval Air Station: 1960.

Naval Air Systems Command, Standard Aircraft Characteristics OV-1B Mohawk. Patuxent River Naval Air Station: 1961.

Naval Air Systems Command, Standard Aircraft Characteristics OV-1C Mohawk. Patuxent River Naval Air Station: 1963.

Naval Air Systems Command, Standard Aircraft Characteristics OV-10D Bronco. Patuxent River Naval Air Station: 1980.

Naval Air Systems Command, Standard Aircraft Characteristics MH-53E Sea Stallion. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics MV-22 Osprey. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics P-3C Orion. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics RA-3B Skywarrior. Patuxent River Naval Air Station: 1959.

Naval Air Systems Command, Standard Aircraft Characteristics S-3A Viking. Patuxent River Naval Air Station: 1973.

Naval Air Systems Command, Standard Aircraft Characteristics SH-3F Sea Sprite. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics SH-3G Sea King. Patuxent River Naval Air Station: 1982.

Naval Air Systems Command, Standard Aircraft Characteristics SH-60B Seahawk. Patuxent River Naval Air Station: 1986.

Naval Air Systems Command, Standard Aircraft Characteristics T-2B Buckeye. Patuxent River Naval Air Station: 1970.

Naval Air Systems Command, Standard Aircraft Characteristics T34C Turbomenter. Patuxent River Naval Air Station: 1978.

Naval Air Systems Command, Standard Aircraft Characteristics T-44A King Air. Patuxent River Naval Air Station: 1978.

Naval Air Systems Command, Standard Aircraft Characteristics T-45A Goshawk. Patuxent River Naval Air Station: 1994.

Naval Air Systems Command, Standard Aircraft Characteristics T-A4F Skyhawk. Patuxent River Naval Air Station: 1964.

Naval Air Systems Command, Standard Aircraft Characteristics TH-57C Jet Ranger. Patuxent River Naval Air Station: 1984.

Naval Air Systems Command, Standard Aircraft Characteristics UC-12B Super King Air. Patuxent River Naval Air Station: 1982.

Naval Air Systems Command, Standard Aircraft Characteristics UH-1N Huey. Patuxent River Naval Air Station: 1982.

Oliver, Dave, Fuel Efficiency as a Key Performance Parameter. Principal Deputy Undersecretary of Defense (AT&L) Memorandum for Director, Strategic and Tactical Systems and Director, Acquisition Resources and Analysis, 15 February 2001.

U.S. Department of Defense, Defense Science Board Report: More Capable Warfighting Through Reduced Fuel Burden. Washington DC: 2001.

U.S. Department of Defense, Joint Strike Fighter Operational Requirements Document. Washington DC: 2000.

U.S. Department of Defense, Military Specification Performance Data Report for Standard Aircraft Characteristics Charts for Piloted Aircraft. MIL-P-2203. Washington DC: 1959.

U.S. Department of Defense, Multi-mission Maritime Aircraft Initial Requirements Document – Release 1. Washington DC: 2001.

U.S. Department of Defense, Operation of the Defense Acquisition System. DoD Directive 5000.1D. Washington DC: 2001.

U.S. Department of the Army, Technical Manual AH-64A Apache (TM 55-1520-238-10). Washington DC: 1989.

U.S. Navy Department, Naval Warfare Publication 3-22.5-F-14A/B/D Volume III Tactical Manual. Washington DC: 2002.

U.S. Naval War College, Resource Allocation: The Formal Process. Newport: 2001.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Professor Lyn Whitaker
Department of Operations Research
Naval Postgraduate School
Monterey, California
4. Professor Paul Sanchez
Department of Operations Research
Naval Postgraduate School
Monterey, California
5. Mr. Jack Keane
Johns Hopkins University Applied Physics Laboratory
Laurel, Maryland
6. Dr. Peter Pandolfini
Johns Hopkins University Applied Physics Laboratory
Laurel, Maryland
7. Mr. Tom Rudowski
Naval Air Systems Command
NAS Patuxent River, Maryland
8. Professor David Schrady
Department of Operations Research
Naval Postgraduate School
Monterey, California
9. LT Larry Young, USN
Silver Spring, Maryland
10. LCDR Arthur Cimiluca, USN
Office of the Chief of Naval Operations
Washington, DC

THIS PAGE INTENTIONALLY LEFT BLANK